



Una solución basada en modelos para la generación de procesos y casos de negocios colaborativos

Memoria de tesis doctoral presentada para la obtención del grado de Doctor por la Universidad de Sevilla

Doctorando:

Miguel Ángel Barcelona Liédana

Dirigido por:

Doctora Dña. María José Escalona Cuaresma
Doctora Dña. Isabel Ramos Román

Sevilla, 16 de noviembre de 2016



Una solución basada en modelos para la generación de procesos y casos de negocios colaborativos

Memoria de tesis doctoral presentada para la obtención del grado de Doctor por la Universidad de Sevilla

Doctorando:

Miguel Ángel Barcelona Liédana

Dirigido por:

Doctora Dña. María José Escalona Cuaresma
Doctora Dña. Isabel Ramos Román

Sevilla, 16 de noviembre de 2016

*A Laura,
por enseñarnos con su esfuerzo
y su eterna sonrisa
que todo es ~~im~~posible*

*El esfuerzo es la magia
que transforma
los éxitos en realidad.*
Diego Pablo Simeone

Agradecimientos

Tengo la enorme fortuna de ponerle cara y nombre a este trabajo, cuyos resultados son fruto de la colaboración y ayuda de muchas personas a las que quiero agradecer haber formado parte de este viaje.

En primer lugar a mis directoras, Isabel y María José, por su dedicación y esfuerzo durante todo este tiempo, por su ánimo constante desde la distancia, por hacerme ver que la meta no está lejana y por todas las mejoras y correcciones realizadas siempre con una sonrisa.

A mis padres, Ángel y María Pilar, que desde pequeño me enseñaron el valor del esfuerzo, de ser una buena persona y una persona buena, con una confianza ciega de que lo lograría. Gracias a ellos sé que puedo afrontar casi cualquier reto y estoy seguro de que hoy, esa estrella que brilla más, representa la sonrisa de un padre orgulloso. Papá, *el chico* lo ha conseguido.

A Laura, principal responsable de este viaje y de tantos otros en los que mi ordenado uso de la razón me dice que dónde vamos. Porque nos enseña cada día a ver del problema una oportunidad, de la locura un reto, del tirar la toalla a seguir remando. Por ponernos en canción, porque aunque no te lo diga ya sabemos que tiene planes para dentro de un año. Por su sonrisa y por demostrarnos que todo es ~~im~~posible.

A esas tres princesas que llenan mi corazón y mi vida, Inés, Rocío y Marina, y que con su mami forman mi CBG (las cuatro Barcelona García) personal. Por todo el tiempo de juegos que este trabajo les quitó. Porque cuando tengan que superarse puedan recordar que, con esfuerzo, serán capaces de todo aquello que se propongan.

A Guillermo, compañero y amigo, por mirar para otro lado cuando creyó que esto no era posible, por saber aprender y saber enseñar, y porque sin quererlo, está en camino y pronto estaremos de nuevo en Sevilla celebrando su tesis.

Al excelente equipo humano del grupo IWT2 del departamento de Lenguajes y Sistemas Informáticos, nuestra familia del sur. Porque nunca dicen no, porque han hecho que Sevilla y Zaragoza estén a un click de distancia y por celebrar cada resultado como propio. Dicen que la memoria guarda los conceptos como imágenes, y a mí me salen todos ellos por defecto cuando pienso en equipo, en compañeros y en buena gente.

A Margaret Ross y Geoff Staples que nos acogieron y cuidaron durante nuestra estancia en Southampton. Porque cada segundo con ellos es un aprendizaje. Porque gracias a Maggie, su *maybe* nos acompaña en cada nuevo reto. A Gustavo Rossi, a Silvia y a todo su equipo del laboratorio LIFIA, por su amabilidad en nuestra visita a La Plata y por su constante voluntad de cooperación.

A José Manuel Cólom y Carlos Ángel Iglesias, que me ayudaron a dar los primeros pasos del camino y me enseñaron que a andar se aprende andando.

A mis amigos, que entre cervezas han intentado entender este trabajo aunque la temática les fuera completamente ajena. *Desde la distancia, qué pequeño todo es.* Ellos han sabido alejarnos cuando ha sido necesario.

Seguro que no he mencionado a muchas personas que han estado ahí y han ayudado, de alguna manera, a llegar a esta cima. A todos ellos, mil gracias. Lo verdaderamente importante no ha sido llegar, sino haber podido disfrutar con vosotros del camino.

Resumen

Hoy en día la mayoría de productos y servicios existentes son fruto de la colaboración de un gran número de compañías que forman una cadena de valor, a veces denominada cadena de suministro (CdS). La globalización y las tecnologías de la información han ayudado a que en la actualidad existan complejas y dinámicas CdS. Aunque cada organización tiene sus propios objetivos e intereses, su capacidad de optimizar las decisiones está cada vez más condicionada por las decisiones que toman aquellas compañías que forman parte de su CdS.

La competencia ya no se establece entre organizaciones individuales sino entre las propias CdS. De esta forma, las organizaciones colaboran para optimizar decisiones que van más allá de las mejoras que una compañía pueda hacer a nivel individual. La gestión individual de los procesos requiere de una visión holística que incorpore una vista inter organizacional como apoyo a la toma de decisiones en la CdS.

Para conseguir eficiencia las organizaciones gestionan sus procesos y crean modelos de los mismos para analizarlos, mejorarlos, optimizarlos, así como para garantizar su uso de forma sistemática e institucionalizada. Existen muchos lenguajes que soportan el modelado de procesos, si bien la mayoría de ellos se centra en la perspectiva individual y no en la colaboración en la CdS. Aquellos que cubren la colaboración plantean un modelo *top-down*, partiendo del diseño del proceso global y posteriormente detallando el comportamiento individual, para lo cual todas las organizaciones deben acordar un único lenguaje tanto para modelar tanto su operativa interna como la colaboración en la CdS, lo que en la práctica impide la reutilización de la perspectiva individual que cada organización tenga de su operativa.

Por ello este trabajo de tesis se centra en la necesidad de crear modelos de procesos de CdS reutilizando los modelos de los procesos que cada organización pueda haber creado, incluso con diversos lenguajes de modelado, incluyendo las capacidades de flexibilidad y adaptación que son propias en los escenarios de colaboración actuales.

En la forma de abordar el problema este trabajo aporta una idea novedosa, en cuanto a la capacidad de resolver el problema con una nueva perspectiva, *bottom-up*, reutilizando los modelos de los procesos que cada organización pueda haber creado con sus lenguajes de modelado, añadiendo a esa vista individual de la colaboración las restricciones a la coreografía de los procesos con el resto de participantes de la CdS. Esta nueva forma de abordar presenta las siguientes ventajas: 1) permite que cada organización utilice el lenguaje de modelado de procesos de negocio más idóneo para sus necesidades; 2) reduce el riesgo de tener incoherencias entre el

modelo individual y el colaborativo; 3) permite que una misma colaboración pueda ser vista de manera diferente por diversos miembros de la CdS; 4) cada compañía decide, para cada colaboración en la que participa, cómo preservar su autonomía en la toma de decisiones y la privacidad en lo relativo a su proceso interno y; 5) facilita el dinamismo de los modelos para colaborar en una nueva CdS o para colaborar con una nueva organización en la cadena.

Nuestra propuesta de solución se basa en un marco de referencia denominado Collaborative Business Generation (CBG), que pretende abordar el modelado de procesos y casos colaborativos con un enfoque bottom-up. Para ello, haciendo uso de nuevos metamodelos, métodos y transformaciones, somos capaces de representar un modelo de proceso de una organización en: 1) una vista colaborativa del proceso, manteniendo la privacidad y la autonomía en la toma de decisiones y; 2) una vista del caso en colaboración, separando la parte estática del posible comportamiento dinámico.

Para alcanzarlo el marco CBG está formado por: 1) el metamodelo CBGProcess que, extiende la propuesta INROMA [García-Borgoñon, 2016] para dar soporte al modelado de procesos de negocio colaborativos; 2) el metamodelo CBGCase que, siguiendo el paradigma de gestión de casos, soporta el modelado de casos en entornos colaborativos; 3) las transformaciones modelo de modelo entre los tres metamodelos anteriores; 4) los métodos para llevar a cabo el modelo de proceso y caso colaborativo a partir de los modelos individuales, y; 5) CBG-Tool, la herramienta para facilitar la creación de modelos y la ejecución de las transformaciones desde un punto de vista práctico y favorecer su uso en entornos empresariales.

La propuesta junto con la herramienta han sido validadas mediante casos de estudio reales extraídos de proyectos de transferencia con empresas en los que se observa cómo la propuesta desarrollada ha sido de gran utilidad en los mismos.

En definitiva, esta tesis doctoral plantea el desarrollo de un marco de referencia, tanto teórico como soportado con herramientas para su uso en la práctica, para facilitar la toma de decisiones en la CdS, mediante el modelado de los procesos colaborativos generados a partir de la vista individual de cada organización y añadiendo la flexibilidad y adaptación dinámica del dominio de casos.

Abstract

Today most existing products and services are the result of the collaboration of a large number of companies that form a value chain, sometimes known as Supply Chain (SC). Globalization and information technologies have helped that currently complex and dynamic SCs exist. Although each organization has its own goals and interests, its ability to optimize decisions is increasingly conditioned by the decisions taken by those companies that are part of its SC.

Competition no longer exists among individual organizations but among SCs. In this way, organizations collaborate to optimize decisions that go beyond single improvements. Individual process management requires a holistic vision that incorporates an inter organizational view that supports SC decision making.

To achieve efficiency organizations manage their processes and create models for analysis, improvement, optimization and to ensure their use in a systematic way. There are many languages that support process modelling, although most of them are focused on the individual perspective rather than SC collaboration. Those which cover collaboration propose a top-down approach, based on the design of the overall process and subsequently detailing individual behavior, so that all organizations must agree on a single language which in practice prevents reuse of the individual perspective.

Therefore this thesis focuses on the need to create SC process models by reusing individual models that each organization may have created, even with different languages, including flexibility and adaptation capabilities that are inherent in collaborative scenarios.

This work provides a novel idea trying to address the problem with a new perspective, a *bottom-up* approach, reusing process models that each organization may have created with their modeling languages. Then adding to that individual view the restrictions into the process choreography according to other supply chain participants. This new approach has the following advantages: 1) allows each organization to use the best process modeling language according to their business needs; 2) reduces the risk of inconsistencies among individual and collaborative models; 3) allows a single collaboration to be seen differently by various supply chain members; 4) each company decides, for all supply chains, how to preserve its autonomy and privacy with regard to its internal process and; 5) provides dynamic models by adding new supply chain collaboration or new participants in the current chain.

Our proposal is the Collaborative Business Generation (CBG) Framework, which aims to address collaborative process and case modelling with a bottom-up approach. To this end, using new meta-models, methods and transformations, we are able to reuse an individual process model to handle: 1) a collaborative process view, maintaining privacy and autonomy in decision-making and; 2) a collaborative case view, separating the static and the dynamic behavior.

CBG framework is composed of: 1) CBGProcess metamodel which extends INROMA to support collaborative business process modelling; 2) CBGCase metamodel that, according case management paradigm, supports collaborative case modelling; 3) model-to-model transformations among the three previous metamodels; 4) methods to carry out the collaborative modelling task, and; 5) CBG-Tool, the support tool to encourage its usage in practice through model editors and model transformations automation.

The proposal and its tool have been validated through real use cases derived from projects with companies. In short, this thesis proposes the development of a framework to facilitate SC decision-making, through the collaborative process modelling from the single view of each organization, and adding flexibility and dynamic adaptation from case domain.

Índice general

Agradecimientos	IV
Resumen	VI
Abstract	VIII
Índice de figuras	XIX
Índice de tablas	XXVII
1. Introducción	1
1.1. La colaboración en la cadena de suministro	3
1.1.1. La cadena de suministro	3
1.1.2. La gestión de la cadena de suministro	4
1.1.3. Modelos y marcos de referencia en la gestión de la cadena de suministro .	11
1.1.4. Estado actual en la gestión de la cadena de suministro	13
1.1.5. Dinamismo, flexibilidad y adaptación en los procesos	15
1.2. Los procesos de negocio en la cadena de valor empresarial	17
1.2.1. Los procesos de negocio en la cadena de suministro	17
1.2.2. Los procesos de negocio colaborativos entre organizaciones	19

1.3. Tecnologías de soporte a la gestión de procesos de negocio	21
1.3.1. Modelado de procesos de negocio	21
1.3.2. La gestión de casos	23
1.3.3. La ingeniería del software dirigida por modelos	29
1.3.4. La computación basada en servicios	33
1.3.5. Tendencias en la aplicación de MDE y SOC para la gestión de procesos de negocio	38
1.4. Problema a resolver	40
1.4.1. Hipótesis	40
1.4.2. Problema a resolver	41
1.4.3. Esquema de la solución	43
1.5. Estructura de la tesis	46
1.6. Conclusiones	48
2. Estado del arte	49
2.1. Metodología para la realización del estado del arte	50
2.1.1. Antecedentes y trabajos relacionados con el estudio de la situación actual	51
2.1.2. Metodologías para desarrollar una revisión sistemática de la literatura . .	58
2.1.3. Preguntas del estudio	59
2.1.4. Estrategia de Búsqueda	60
2.1.5. Criterios de inclusión y exclusión	62
2.1.6. Aseguramiento de la calidad	65
2.1.7. Recogida y análisis de información	67
2.2. Resultados del estudio del estado del arte	68

2.2.1. Resultados de la búsqueda	68
2.2.2. Evaluación de la calidad	69
2.3. Análisis del estado del arte	71
2.3.1. RQ1: ¿Qué lenguajes de modelado de procesos de negocio de la CdS se han definido?	71
2.3.2. RQ2: ¿Qué problemas de la CdS se han intentado resolver?	79
2.3.3. RQ3: ¿Cuáles son las limitaciones de la investigación actual?	81
2.3.4. Limitaciones del estudio	82
2.4. Oportunidades para la futura investigación	84
2.5. Conclusiones	86
3. Planteamiento del Problema	89
3.1. Aspectos relevantes que determinan el problema a resolver	90
3.1.1. Las organizaciones necesitan definir y mejorar los procesos de su cadena de valor	90
3.1.2. La mejora de los procesos de la CdS requiere de entornos basados en modelos dinámicos, flexibles y adaptables	91
3.1.3. Existen muchos lenguajes de modelado de procesos de negocio y de la CdS	92
3.2. Planteamiento del problema a resolver	93
3.2.1. Necesidad a resolver	93
3.2.2. Casos de uso	94
3.2.3. Marco teórico	96
3.3. Objetivos a alcanzar	98
3.4. Influencias conceptuales y tecnológicas	99
3.4.1. Las lecciones aprendidas de la revisión sistemática de la literatura	99

3.4.2. La ingeniería del software basada en modelos	100
3.4.3. La gestión de casos	103
3.4.4. INROMA: un marco de referencia para facilitar la interoperabilidad y mantenibilidad de los modelos de procesos	105
3.5. Estructura de la solución	109
3.6. Conclusiones	117
4. Lenguaje y método para el modelado de procesos colaborativos a partir de modelos de procesos individuales	119
4.1. Necesidades que justifican la extensión de INROMA	121
4.2. Metamodelo <i>CBGProcess</i>	123
4.2.1. Activity	126
4.2.2. BlackBoxActivity	126
4.2.3. BlackBoxIndicator	127
4.2.4. BlackBoxMetric	127
4.2.5. BlackBoxProduct	128
4.2.6. Conditional	129
4.2.7. Deliverable	129
4.2.8. Final	130
4.2.9. FlowElement	131
4.2.10. Indicator	131
4.2.11. Initial	132
4.2.12. Metric	132
4.2.13. Organization	133
4.2.14. Process	133

4.2.15. ProcessElement	134
4.2.16. ProcessSequence	135
4.2.17. Product	135
4.2.18. Stakeholder	136
4.3. Método para definir el proceso colaborativo	137
4.4. Conclusiones	139
5. Lenguaje y método para el modelado de casos colaborativos	141
5.1. Lenguajes para modelar casos colaborativos	142
5.2. Metamodelo <i>CBGCCase</i>	144
5.2.1. Case	145
5.2.2. CaseData	146
5.2.3. CaseElement	146
5.2.4. CaseFlowElement	147
5.2.5. CaseSequence	147
5.2.6. CaseTask	148
5.2.7. Constraint	148
5.2.8. Conditional	149
5.2.9. DiscretionarySequence	150
5.2.10. Final	150
5.2.11. HumanTask	151
5.2.12. Initial	151
5.2.13. MandatorySequence	152
5.2.14. Organization	152

5.2.15. Plan	153
5.2.16. ProcessTask	153
5.2.17. Role	154
5.3. Método para definir casos colaborativos	155
5.4. Conclusiones	158
6. Transformaciones para la generación de procesos y casos colaborativos	159
6.1. Selecccion de QVT para describir las transformaciones entre modelos	161
6.2. Transformaciones en el marco CBG	163
6.2.1. Transformación M2M horizontal para generar la vista colaborativa de un proceso individual	164
6.2.2. Transformación M2M de procesos a casos colaborativos	166
6.3. Conclusiones	169
7. CBG-Tool: Herramienta de soporte para la generación de procesos y casos de negocios colaborativos	171
7.1. Casos de uso, entorno de trabajo y arquitectura de CBG-Tool	173
7.1.1. Casos de uso	173
7.1.2. Entorno de trabajo de CBG-Tool: Enterprise Architect	175
7.1.3. Arquitectura	177
7.2. La herramienta CBG-Tool	178
7.2.1. El lenguaje de modelado de procesos colaborativos CBGProcess en CBG- Tool	178
7.2.2. El lenguaje de modelado de procesos colaborativos CBGCase en CBG-Tool	182
7.2.3. Implementación de las Transformación de INROMA al proceso colabora- tivo en CBG-Tool	186

7.2.4. Implementación de las Transformación del proceso al caso colaborativo en CBG-Tool	187
7.3. Validación de CBG-Tool	189
7.3.1. Crear la vista colaborativa a partir de un modelo de proceso individual	189
7.3.2. Incorporar nuevos procesos al modelo de proceso colaborativo	194
7.3.3. Un modelo de proceso individual en varios procesos colaborativos	197
7.3.4. Crear el modelo del caso colaborativo	199
7.4. Conclusiones	201
8. Evaluación del marco CBG en entornos reales: caso de estudio	203
8.1. Elección del caso de estudio	204
8.2. Caso de estudio: ITCHAIN	206
8.2.1. Obtener la descripción de los procesos individuales	209
8.2.2. Establecer la vista colaborativa del proceso individual	218
8.2.3. Construir el modelo de proceso colaborativo	224
8.2.4. Construir el modelo de caso colaborativo	226
8.3. Conclusiones	228
9. Conclusiones y trabajos futuros	229
9.1. Marco de investigación en el que se desarrolla este trabajo	231
9.1.1. Línea de investigación en el grupo IWT2	231
9.1.2. Línea de investigación en ITAINNOVA	232
9.1.3. Relaciones con otros organismos que han influido en el desarrollo de este trabajo de tesis	232
9.2. Aportaciones de esta tesis	235

9.2.1. Un estudio del estado del arte sobre lenguajes de modelado de procesos de la CdS	235
9.2.2. Un lenguaje de modelado de procesos de negocio colaborativos	236
9.2.3. Un método para construir el modelo de proceso colaborativo	236
9.2.4. Un lenguaje de modelado de casos colaborativos	237
9.2.5. Una transformación para convertir los modelos de procesos de negocio colaborativos en modelos de casos colaborativos	237
9.2.6. Una herramienta de soporte para su aplicación práctica	238
9.2.7. Validación de la propuesta con casos reales	238
9.2.8. Aportaciones no tecnológicas	240
9.3. Trabajos futuros	241
9.3.1. La capacidad de simular modelos de procesos y casos colaborativos	241
9.3.2. La capacidad de ejecutar procesos y casos colaborativos	242
9.3.3. La capacidad de orientar el modelo a procesos intensivos en conocimiento	242
9.3.4. La capacidad de incorporar el factor humano en el modelo	243
9.3.5. La definición de un mecanismo que facilite las auditorias y certificaciones de calidad de los procesos	243
9.4. Conclusiones	244
Bibliografía	244
A. Transformaciones M2M entre INROMA y CBGProcess	263
B. Transformaciones M2M entre CBGProcess y CBGCase	277
C. Manual de usuario de CBG-Tool	293
C.1. Instalación del entorno de la herramienta	294

C.2. Crear la vista individual del modelo de proceso colaborativo	296
C.3. Crear el modelo de proceso colaborativo	298
C.4. Crear el modelo de caso colaborativo	299
D. Actividad investigadora	303
D.1. Producción investigadora	304
D.1.1. Capítulos de libro	304
D.1.2. Revistas	304
D.1.3. Conferencias internacionales	305
D.1.4. Conferencias nacionales	306
D.2. Actividad investigadora	307
D.2.1. Estancias de investigación	307
D.2.2. Comités	307
D.2.3. Redes de investigación	308
D.2.4. Proyectos	308
Acrónimos	313
Glosario de Términos	317

Índice de figuras

1.1. Formas de ver las diferencias entre SCM y Logística	6
1.2. Clasificación de trabajos y participación de las personas, adaptado de [Breitenmoser y Keller, 2015]	15
1.3. Metamodelo de gestión de casos propuesto por Van der Aalst et al. [Van der Aalst <i>et al.</i> , 2005]	24
1.4. Características de los Adaptive Case Management [Herrmann y Kurz, 2011] . . .	24
1.5. Fases de diseño y planificación-ejecución del modelo CMMN	26
1.6. Principales elementos del metamodelo de CMMN (elaboración propia desde [OMG, 2014])	27
1.7. Transformaciones M2M y M2T	31
1.8. Transformaciones M2M endógenas y exógenas	32
1.9. Dimensiones ortogonales de las transformaciones M2M con ejemplos	33
1.10. Enfoques top-down y bottom-up para el modelado de la CdS	42
1.11. Resumen de la estructura de solución planteada en el marco CBG	44
2.1. Fases principales de la metodología de la revisión sistemática de la literatura . .	50
2.2. Fases principales del proceso de búsqueda de estudios primarios	60
2.3. Proceso para seleccionar los estudios incluidos en la búsqueda inicial	63
2.4. Proceso para seleccionar los estudios incluidos para crear la bola de nieve	64

2.5. Proceso para discernir si un estudio es incluido a partir del texto completo	64
2.6. Número de estudios por fase	68
2.7. Porcentaje de estudios incluidos por fase	69
2.8. Evaluación de cada pregunta de aseguramiento de calidad en base a los estudios incluidos	70
2.9. Resumen de las características analizadas en la RQ1	77
2.10. Enfoque de modelado de las propuestas existentes	78
3.1. Planteamiento teórico del problema a resolver	96
3.2. Niveles MOF de las coreografías de procesos [Weske, 2010]	100
3.3. Los lenguajes de QVT: relaciones y arquitectura	101
3.4. Sintaxis concreta gráfica de QVT Relations	103
3.5. Detalle de metamodelos y transformaciones de INROMA	105
3.6. Vista del marco de referencia propuesto en INROMA	106
3.7. Vista del contexto del marco de referencia INROMA	107
3.8. Metamodelo de interoperabilidad de lenguajes de procesos software	108
3.9. Resumen de la estructura de solución planteada en el marco CBG	110
3.10. Vista del primer paso del marco CGB	111
3.11. Vista del segundo paso del marco CGB	112
3.12. Vista del tercer paso del marco CGB	113
3.13. Proceso general del marco CGB	115
4.1. Uso del mecanismo de extensibilidad de UML sobre INROMA	122
4.2. Metamodelo <i>CBGProcess</i> para el modelado de procesos colaborativos	124
4.3. Método para definir el proceso colaborativo	138

5.1. Metamodelo para la generación de casos colaborativos	144
5.2. Método para definir el caso colaborativo	156
6.1. Transformaciones M2M en el marco CBG	160
6.2. Proceso de generación de transformaciones en el marco CBG	163
6.3. Transformación M2M INROMA2CBGProcess en el marco CBG	164
6.4. Vista QVT Relations de la relación Process2Process en la transformación M2M inroma2CBGProcess	165
6.5. Transformación M2M CBGProcess2CBGCase en el marco CBG	166
6.6. Vista QVT Relations de la relación Process2Case en la transformación M2M CBGProcess2CBGCase	166
6.7. Vista QVT Relations de la relación MapActivities en la transformación M2M CBGProcess2CBGCase	167
6.8. Vista QVT Relations de la relación MapInputProductsOfActivity en la transfor- mación M2M CBGProcess2CBGCase	168
7.1. Casos de uso de la herramienta CBG-Tool	173
7.2. Arquitectura de extensión de componentes de Enterprise Architect	176
7.3. Arquitectura de la herraminta CBG-Tool dentro de Enterprise Architect	177
7.4. Arquitectura de la herraminta CBG-Tool dentro del marco CBG	177
7.5. Perfil UML para CBGProcess	178
7.6. Sintaxis concreta de CBGProcess en CBG-Tool	180
7.7. Editor de CBGProcess en CBG-Tool	181
7.8. Perfil UML para CBGProcess	182
7.9. Sintaxis concreta de CBGCase en CBG-Tool	184
7.10. Editor de CBGCase en CBG-Tool	185

7.11. Vista QVT Relations de la relación Process2Process en la transformación M2M inroma2CBGProcess	186
7.12. Vista QVT Relations de la relación Process2Case en la transformación M2M CBGProcess2CBGCase	187
7.13. Modelo de proceso individual de la Organización 1	190
7.14. Modelo de proceso individual de la Organización 2	190
7.15. Vista individual de la Organización 1 sobre el proceso en colaboración	191
7.16. Vista individual de la Organización 2 sobre el proceso en colaboración	192
7.17. Modelo de proceso colaborativo entre las Organizaciones 1 y 2	193
7.18. Modelo de proceso individual de una nueva Organización 3	194
7.19. Vista individual de la Organización 3 sobre el proceso en colaboración	195
7.20. Modelo de proceso colaborativo entre las Organizaciones 1, 2 y 3	196
7.21. Modelo de proceso colaborativo entre las Organizaciones 4 y 5	197
7.22. Vista individual de la Organización 1 sobre el proceso en colaboración, tanto en la cadena 1 y 2 como con las organizaciones 4 y 5	198
7.23. Modelo de caso colaborativo entre las Organizaciones 1, 2 y 3	199
7.24. Vista individual de la Organización 1 sobre el caso en colaboración entras las organizaciones 1, 2 y 3	200
8.1. Contexto general del modelado del proceso colaborativo de compra y distribución de pescado fresco	207
8.2. Vistas del modelo de proceso individual y en colaboración obtenidas en ITChain	208
8.3. Modelo de proceso colaborativo obtenido en ITChain	208
8.4. Modelo de proceso individual en INROMA de la compra en la lonja	211
8.5. Modelo de proceso individual en INROMA de la coordinación de la compra . . .	213
8.6. Modelo de proceso individual en INROMA de un cambio en la planificación logísti- ca (almacén)	215

8.7. Modelo de proceso individual en INROMA de un cambio en la planificación del transporte	217
8.8. Modelo de la vista individual del proceso colaborativo desde el comprador en la lonja	219
8.9. Modelo de la vista individual del proceso colaborativo desde el coordinador de compras	221
8.10. Modelo de la vista individual del proceso colaborativo desde el responsable de almacén	222
8.11. Modelo de la vista individual del proceso colaborativo desde el responsable de transporte	223
8.12. Modelo del proceso colaborativo de aprovisionamiento, transporte y distribución	225
8.13. Modelo del caso colaborativo de aprovisionamiento, transporte y distribución . .	227
A.1. Vista QVT Relations de la relación Process2Process en la transformación M2M inroma2CBGProcess	264
A.2. Vista QVT Relations de la relación MapActivities en la transformación M2M inroma2CBGProcess	264
A.3. Vista QVT Relations de la relación MapIndicators en la transformación M2M inroma2CBGProcess	265
A.4. Vista QVT Relations de la relación MapInitial en la transformación M2M inroma2CBGProcess	265
A.5. Vista QVT Relations de la relación MapFinal en la transformación M2M inroma2CBGProcess	266
A.6. Vista QVT Relations de la relación MapConditional en la transformación M2M inroma2CBGProcess	266
A.7. Vista QVT Relations de la relación MapInputProductsOfActivity en la transformación M2M inroma2CBGProcess	267
A.8. Vista QVT Relations de la relación MapOutputProductsOfActivity en la transformación M2M inroma2CBGProcess	267
A.9. Vista QVT Relations de la relación MapStakeholdersPartOfActivity en la transformación M2M inroma2CBGProcess	268

A.10. Vista QVT Relations de la relación MapStakeholdersRespOfActivity en la transformación M2M inroma2CBGProcess	268
A.11. Vista QVT Relations de la relación MapIncomingSequence en la transformación M2M inroma2CBGProcess	268
A.12. Vista QVT Relations de la relación MapOutgoingSequence en la transformación M2M inroma2CBGProcess	269
A.13. Vista QVT Relations de la relación MapMetrics en la transformación M2M inroma2CBGProcess	269
B.1. Vista QVT Relations de la relación Process2Case en la transformación M2M CBGProcess2CBGCase	278
B.2. Vista QVT Relations de la relación MapOrganizations en la transformación M2M CBGProcess2CBGCase	278
B.3. Vista QVT Relations de la relación MapActivities en la transformación M2M CBGProcess2CBGCase	279
B.4. Vista QVT Relations de la relación MapInitial en la transformación M2M CBGProcess2CBGCase	279
B.5. Vista QVT Relations de la relación MapFinal en la transformación M2M CBGProcess2CBGCase	280
B.6. Vista QVT Relations de la relación MapConditional en la transformación M2M CBGProcess2CBGCase	280
B.7. Vista QVT Relations de la relación MapStakeholders en la transformación M2M CBGProcess2CBGCase	280
B.8. Vista QVT Relations de la relación MapInputProductsOfActivity en la transformación M2M CBGProcess2CBGCase	281
B.9. Vista QVT Relations de la relación MapOutputProductsOfActivity en la transformación M2M CBGProcess2CBGCase	281
B.10. Vista QVT Relations de la relación MapIncludedInputProducts en la transformación M2M CBGProcess2CBGCase	282
B.11. Vista QVT Relations de la relación MapIncludedOutputProducts en la transformación M2M CBGProcess2CBGCase	282

B.12. Vista QVT Relations de la relación MapIncludedActivitiesOfActivity en la transformación M2M CBGProcess2CBGCase	282
B.13. Vista QVT Relations de la relación MapStakeholdersPartOfActivity en la transformación M2M CBGProcess2CBGCase	283
B.14. Vista QVT Relations de la relación MapStakeholdersRespOfActivity en la transformación M2M CBGProcess2CBGCase	283
B.15. Vista QVT Relations de la relación MapIncomingSequence en la transformación M2M CBGProcess2CBGCase	283
B.16. Vista QVT Relations de la relación MapOutgoingSequence en la transformación M2M CBGProcess2CBGCase	284
C.1. Editores de INROMA, CBGProcess y CBGCase mediante perfiles UML desplegados en Enterprise Architect	294
C.2. Addin de CBGTool desplegado en Enterprise Architect mediante un menu contextual	295
C.3. Uso de CBGTool para transformar el modelo de proceso INROMA a CBGProcess	296
C.4. Uso de CBGTool para crear la vista individual del proceso colaborativo	297
C.5. Uso de CBGTool para crear el modelo de proceso colaborativo	298
C.6. Uso de CBGTool para transformar el modelo de proceso CBGProcess a CBGCase	299
C.7. Uso de CBGTool para crear el modelo de caso colaborativo	300
C.8. Uso de CBGTool para crear la vista individual del modelo de caso colaborativo .	301

Índice de tablas

1.1. Diferencias y similitudes entre BPM y SOA	35
1.2. Características de SOA con potencial aplicación a un entorno inter organizacional	37
2.1. Diversas palabras de búsqueda que fueron utilizadas en la primera fase	61
2.2. Librerías digitales y criterios para realizar la búsqueda	61
2.3. Criterios de inclusión y exclusión en la fase de búsqueda	62
2.4. Preguntas para evaluar la calidad de los estudios incluidos	66
2.5. Esquema de información recogida para cada estudio incluido	67
2.6. Propuestas de lenguajes de modelado de procesos de la CdS existentes	72
2.7. Propuestas de lenguajes de modelado de procesos de la CdS existentes (cont) . .	73
2.8. Propuestas de lenguajes de modelado de procesos de la CdS existentes (cont) . .	74
2.9. Análisis de las propuestas de lenguajes de modelado de procesos de la CdS existentes	76
2.10. Principales problemas de la CdS que han intentado resolver las propuestas existentes	80
2.11. Términos por los que se conoce al modelado de procesos de negocio de la CdS . .	80
2.12. Líneas de trabajo futuro identificadas en las propuestas	81
2.13. Líneas de trabajo futuro identificadas en otras revisiones de la literatura	85
3.1. Relación entre los objetivos planteados y la estructura de la solución propuesta .	116

6.1. Relaciones para acometer la transformación INROMA2CBGProcess	165
6.2. Relaciones para acometer la transformación CBGProcess2CBGCase	168
7.1. Descripción del caso de uso Crear vista individual del modelo de proceso colaborativo	174
7.2. Descripción del caso de uso Crear modelo de proceso colaborativo	174
7.3. Descripción del caso de uso Crear modelo de caso colaborativo	175
7.4. Justificación de las metaclasses en el perfil UML de CBGProcess	179
7.5. Justificación de las metaclasses en el perfil UML de CBGCase	183
8.1. Información detallada del proyecto ITChain	205
A.1. Relaciones para acometer la transformación INROMA2CBGProcess	263
B.1. Relaciones para acometer la transformación CBGProcess2CBGCase	277

Capítulo 1

Introducción

Hoy en día la mayoría de productos existentes son fruto de la colaboración de un gran número de compañías diferentes que forman una cadena de valor desde las materias primas hasta el cliente final. De igual forma ocurre con la prestación de servicios, en la que un conjunto de organizaciones cooperan para ofrecer un resultado final al cliente manteniendo unos niveles de servicio preestablecidos. Cada organización que forma parte de la cadena ofrece, de alguna u otra forma, valor al cliente final, que se materializa como resultado de la colaboración efectiva de los miembros de la cadena.

Tradicionalmente se ha denominado cadena de suministro a esa red de proveedores que es necesaria para satisfacer las expectativas del cliente, si bien con este enfoque orientado a ofrecer valor, es también conocida como cadena de valor. Ser capaces de ofrecer más valor al cliente de manera más eficiente es un reto constante en las organizaciones actuales. La globalización, entendida como la capacidad de encontrar proveedores que cubran necesidades en cualquier parte del globo, y las tecnologías de la información, como mecanismo para favorecer de una forma efectiva esa colaboración, han conllevado que en la actualidad existan complejas y dinámicas cadenas de suministro para la mayoría de los productos o servicios existentes. Por ello, aunque cada organización tiene sus propios objetivos e intereses, su capacidad de optimizar las decisiones está cada vez más condicionada por las decisiones que toman aquellas compañías que forman parte de su cadena de valor.

La competencia ya no se establece entre organizaciones individuales sino entre las propias cadenas de valor. De esta forma, el intercambio de información operativa, táctica y estratégica entre las organizaciones para mejorar las decisiones que toman desde su planificación hasta su ejecución, hace que se puedan optimizar decisiones que van más allá de las mejoras que una compañía pueda hacer a nivel individual. Las organizaciones adoptan una gestión basada en procesos con el fin de optimizar sus operaciones y su toma de decisiones. En este contexto, es preciso por tanto abordar una forma de trabajo colaborativa que permita definir los procesos con un enfoque inter organizacional para alinear esa toma de decisiones para maximizar la eficiencia de toda la cadena.

El objetivo de este capítulo es establecer el contexto en el que se ha llevado a cabo el trabajo de tesis. Para ello, en la primera sección se introducen los principales conceptos derivados de la colaboración en una cadena de suministro, intentando enunciar las principales áreas de mejora que justifican nuestro trabajo. En la segunda sección se expone la necesidad de describir los procesos en las organizaciones y, en particular, las relaciones existentes a nivel inter organizacional entre esos procesos de negocio con un enfoque de cadena de valor. Seguidamente, la tercera sección muestra las principales tecnologías de soporte que serán de utilidad para abordar la gestión de procesos de negocio colaborativos. A continuación, la cuarta sección plantea el objetivo de este trabajo de tesis. Seguidamente la quinta sección describe la estructura del presente documento para finalizar con unas breves conclusiones de este capítulo.

1.1. La colaboración en la cadena de suministro

En esta sección vamos a definir los principales conceptos existentes en el dominio de aplicación del presente trabajo de tesis, como son la Cadena de Suministro (CdS) ó *Supply Chain* (SC), la Gestión de la Cadena de Suministro (*Supply Chain Management*, *SCM*), su relación con la logística, los términos de integración y colaboración así como los principales modelos y marcos de referencia existentes en SCM. Por último enumeraremos el punto de partida en la gestión de la cadena de suministro en la actualidad y definiremos las características de dinamismo, flexibilidad y adaptación que concretan el tipo de CdS a las que queremos dar soporte en este trabajo.

1.1.1. La cadena de suministro

El Council of Supply Chain Management Professionals (CSCMP)¹, la primera asociación mundial de profesionales de la CdS, define en su glosario de términos [Council of Supply Chain Management Professionals, 2013] la CdS como:

- Comenzando con las materias primas y finalizando con el uso por parte del cliente final del producto terminado, la CdS son los enlaces que unen las organizaciones que participan.
- El intercambio de materiales e información en los procesos logísticos que va desde la adquisición de materias primas hasta la entrega de productos terminados al usuario final.

Todos los proveedores, prestadores de servicios y los clientes son los eslabones de la CdS. Chopra y Meindl [Chopra y Meindl, 2007] la entienden como todas las partes involucradas, de forma directa o indirecta, para satisfacer una solicitud de un cliente, a la vez que defienden su dinamismo en el continuo flujo de información, producto y finanzas entre los participantes. El término CdS ha sido definido ampliamente en la literatura desde hace muchos años [Janvier-James, 2012].

En esas definiciones se han incluido visiones basadas, entre otras, en:

- El paso y la manipulación de bienes desde las materias primas al producto terminado.
- El intercambio y la distribución de productos hasta el cliente.
- El alineamiento inverso y directo de distintas compañías para ofrecer valor al cliente.

Considerando algunas de estas definiciones, Mentzer et al. [Mentzer *et al.*, 2001] definieron una CdS como el conjunto de tres o más entidades (tanto organizaciones como personas), directamente involucradas en el flujo directo o inverso de productos, servicios, finanzas y/o información desde un origen hasta el cliente final.

¹<http://www.cscmp.org/>, <http://www.cscmpspain.org/>

1.1.2. La gestión de la cadena de suministro

Desde que en 1982 Oliver y Webber empezaron a hablar del término gestión de la cadena de suministro [Oliver y Webber, 1982], ha habido un gran interés tanto a nivel académico como en la industria. La primera revisión sobre el término SCM en 1997 ya identificó que no existía una visión compartida de lo que SCM era o debiera ser, si bien se apuntó a la necesidad de una integración de las operaciones del negocio en la CdS que fuera más allá de la logística. Por ello, Cooper et al. [Cooper *et al.*, 1997] ya definieron SCM como la integración de todos los procesos de negocio clave a lo largo de la CdS.

Es importante resaltar las diferencias existentes entre la CdS y la SCM. Una CdS existe, con independencia de que sea o no gestionada, y las relaciones de negocio entre las organizaciones se llevan a cabo, exista o no una SCM. De esta forma la CdS es un fenómeno existente en el mundo empresarial, algo que en ocasiones se denomina canales de distribución, y SCM es la labor de gestionar la CdS, requiriendo esfuerzos en las organizaciones que la forman [Mentzer *et al.*, 2001].

A pesar de su popularidad e importancia, los conceptos asociados a SCM mantienen una serie de problemas, como: falta de una definición universalmente aceptada de SCM, la existencia de distintos e incluso contradictorios términos así como la relativa falta de evidencias empíricas de que confirmen los beneficios teóricos atribuidos a la SCM [Naslund y Williamson, 2010]. Seguidamente vamos a presentar las principales definiciones existentes en la literatura sobre el término SCM.

El Institute for Supply Management (ISM)², fundado en 1915, define SCM como la identificación, adquisición, acceso, posicionamiento y gestión de los recursos y las capacidades relacionadas que una organización necesita o puede potencialmente necesitar para lograr sus objetivos estratégicos [Institute for Supply Management, 2009].

El SCM Institute (SCMi)³ define SCM como la gestión de las relaciones en la red de organizaciones, desde los clientes finales a través de los proveedores primarios, utilizando procesos de negocio interfuncionales clave para crear valor para los clientes y otras partes interesadas [Supply Chain Management Institute, 2013].

El CSCMP entiende que SCM abarca la planificación y gestión de todas las actividades involucradas en aprovisionamiento y adquisición, conversión, y todas las actividades de gestión de la logística [Council of Supply Chain Management Professionals, 2013]. Es importante destacar que, también incluye la coordinación y la colaboración con otros socios, ya sean proveedores, intermediarios, terceras partes que ofrecen servicios como proveedores y clientes. En esencia, SCM integra la gestión de la oferta y la demanda dentro y a través de las empresas. SCM es una función de integración cuya responsabilidad fundamental se centra en vincular los principales procesos y funciones del negocio entre todas las organizaciones de forma coherente y buscando la eficiencia y el rendimiento del modelo de negocio en su conjunto. Incluye todas las actividades de gestión de la logística anteriormente señaladas, así como la fabricación y dirige la coordinación de

²<http://www.ism.ws>

³<http://scm-institute.org>

los procesos y las actividades con y a través de marketing, ventas, diseño de producto, finanzas y tecnologías de la información [Council of Supply Chain Management Professionals, 2013].

APICS Supply Chain Council (SCC)⁴, organización internacional propietaria del modelo Supply Chain Operations Reference (SCOR), define SCM en su diccionario [APICS Supply Chain Council, 2013] como el diseño, planificación, ejecución, control y seguimiento de las actividades de la CdS con el objetivo de crear valor en la red, construir una infraestructura competitiva, aprovechar una logística globalizada, sincronizar oferta y demanda y medir la eficacia a nivel global.

En 2009 Stock et al. [Stock y Boyer, 2009] analizaron 173 definiciones de SCM en la literatura para, además de constatar el gran número de definiciones existentes y analizar el significativo impacto negativo que esta falta de consenso tiene en la industria y en la comunidad científica, terminar concretando su propia definición de SCM como: La gestión de una red de relaciones dentro de una empresa y entre organizaciones inter dependientes y unidades de negocio que consisten en proveedores de materiales, compras, instalaciones de producción, logística, marketing y los sistemas relacionados que faciliten el flujo directo e inverso de los materiales, servicios, finanzas y la información asociada desde el productor inicial hasta el cliente final con los beneficios de aportar un valor añadido, maximizar la rentabilidad a través de la eficiencia y obteniendo la satisfacción del cliente.

A partir de la revisión de la literatura relacionada con SCM que será expuesta en el capítulo 2, así como de las diversas definiciones y marcos de SCM, existen un conjunto de conceptos claves ampliamente repetidos, entre los que además de la coordinación de los flujos y actividades, destacan la relación entre SCM y logística, así como la colaboración e integración en la SCM. Se hace preciso por tanto exponer con mayor detalle estos términos para comprender sus adopciones y su interpretación en el ámbito de las CdS.

Por logística, en su contexto industrial, APICS la define como el arte y la ciencia de obtener, producir y distribuir materiales y productos terminados en el lugar y cantidad adecuados [APICS Supply Chain Council, 2013]. Dado que en ocasiones se interpreta la SCM como algo exclusivo del dominio de la logística, es decir, de producir y distribuir mercancías o productos, en la siguiente sección profundizamos de cara a establecer la adopción que vamos a seguir en esta tesis.

⁴<https://supply-chain.org>

1. Gestión de la cadena de suministro vs logística

En los últimos años ha habido mucho debate sobre las diferencias existentes entre logística y gestión de la cadena de suministro. Larson y Halldorsson [Larson y Halldorsson, 2004] en el 2004 identificaron cuatro formas de verlo, como se muestra en la figura 1.1. En ella están:

- Los tradicionales, en los que SCM es simplemente una parte de la Logística.
- Los re-etiquetados, que bajo el término SCM engloban lo que antes era Logística.
- Los unionistas, que ven la Logística como una parte dentro de la SCM.
- Los interseccionistas, que sugieren que SCM no es simplemente la unión de tareas logísticas, marketing, gestión de operaciones, compras y el resto de áreas funcionales. Éstos ven que existe una intersección en la que SCM coordina a nivel inter organizacional todos los esfuerzos que cada compañía hace a nivel individual, defendiendo que SCM es algo estratégico, no táctico.

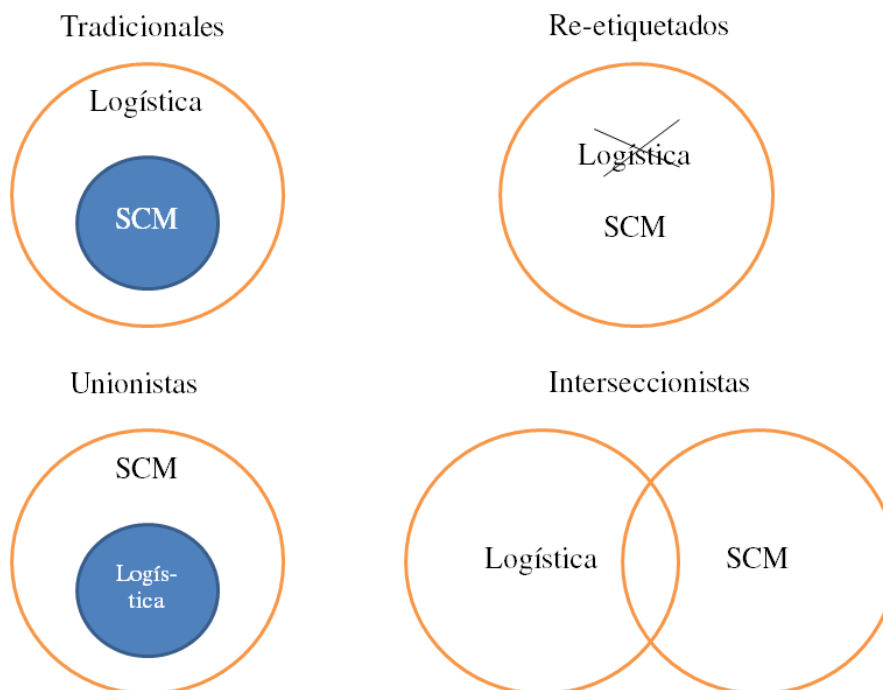


Figura 1.1: Formas de ver las diferencias entre SCM y Logística

En 2005 Sweeney [Sweeney, 2005] analizó la literatura concluyendo que el enfoque unionista, en el que Logística es sólo una parte de SCM, era el más utilizado hasta la fecha. Esta visión defiende que Logística es generalmente concebida dentro de una organización, aunque se incluye la gestión de los flujos con sus proveedores y clientes. SCM incluye todos los flujos logísticos, así como la gestión de las órdenes de pedidos de los clientes, los procesos de fabricación y todos los flujos de información asociados para dar seguimiento a las actividades a lo largo de toda la CdS.

En 2010 Ivanov y Sokolov identificaron doce malos entendidos en el ámbito de la SCM, siendo uno de ellos el hecho de que SCM reemplazara la logística. Como conclusión de su revisión de la literatura, concluyeron que logística hace referencia en la mayoría de las ocasiones a operaciones que realiza una organización para llevar a cabo la transición física de las mercancías, mientras que SCM se asocia con el valor añadido generado a través de la CdS global enfocándose a las conexiones entre esas transiciones en las diversas organizaciones de la CdS. Mientras que logística es aprovisionar las mercancías, en la cantidad, lugar y tiempo adecuados, con los mejores costes, SCM se ocupa de balancear oferta y demanda a lo largo de la CdS para obtener la satisfacción del cliente [Ivanov y Sokolov, 2010].

Atendiendo a esta clasificación, en el presente trabajo de tesis seguiremos esta perspectiva unionista en la que Logística será un dominio de aplicación particular dentro de una SCM más generalista. Por ello, cuando hagamos referencia a SCM estaremos hablando de entornos en los que varias organizaciones deben cooperar para satisfacer un objetivo común, bien sean distribuidores minoristas, empresas TIC o cualquier otro sector industrial. Cabe destacar que cuando estamos hablando de organizaciones distintas en una cadena de valor, bien podríamos estar en el caso de grandes organizaciones que tienen áreas funcionales en su estructura, como pueden ser departamentos de compras, producción y logística, encargados desde el aprovisionamiento hasta la entrega de los productos al cliente. Dado que en la literatura está ampliamente definida la CdS como la participación de dos o más entidades independientes, entendidas como organizaciones jurídicamente distintas, mantendremos esta suposición si bien entendemos que el problema y la solución propuesta podrían ser de aplicación en el contexto de diversas áreas funcionales bajo una misma empresa.

2. Colaboración en la CdS

La colaboración en la CdS se ha convertido en un término habitual para describir los esfuerzos necesarios para conseguir competitividad a largo plazo en una CdS. Por ello en esta sección intentaremos definir el término, su posible alcance en la CdS, sus potenciales ventajas, su adopción en la práctica en la industria y cómo entenderemos este término en nuestro trabajo.

En primer lugar, intentemos definir el concepto de colaboración en la SCM. Para Angerhofer y Angelides [Angerhofer y Angelides, 2006], el objetivo de una CdS colaborativa es ganar ventaja competitiva mejorando el desempeño general de la cadena a través de un enfoque holístico, en lugar de hacerlo a través de la mejora de cada participante de manera independiente. Para Holweg et al. se entiende la colaboración bajo la creencia de que dará lugar a una CdS sincronizada y sin fisuras, que a su vez dará lugar a un mejor servicio al cliente, con menor costes y mayores beneficios [Holweg *et al.*, 2005]. En su forma más avanzada, la colaboración puede incluir la formación de asociaciones y vínculos que permiten a las empresas que forman una CdS trabajar activamente juntas, compartiendo información, conocimiento, riesgos y beneficios con una meta común [Gruat La Forme *et al.*, 2007]. A pesar de existir cuantiosas adopciones, la colaboración en la CdS es todavía un término mal definido que puede incluir desde mayor intercambio de información en el nivel operativo entre dos empresas para obtener una previsión compartida hasta la toma de decisiones conjunta a nivel táctico o estratégico [Ajmera y Cook, 2009].

En relación al alcance de una colaboración, pueden existir formas de colaboración al menos [Angerhofer y Angelides, 2006]:

1. A nivel operativo, en la que la colaboración se centra en las actividades transaccionales diarias.
2. A nivel táctico, en el que se extiende a las actividades para controlar y coordinar el flujo de información y de mercancías.
3. A nivel estratégico, en el que se incluirían las actividades para influir en la futura dirección de la CdS de forma colaborativa, con responsabilidades compartidas entre los actores clave.

Varios autores han analizado las ventajas existentes derivadas de la colaboración en la CdS, entre los que destacan una mayor flexibilidad, una mejor utilización de los recursos, un menor tiempo de entrega o en su defecto mejor control de los retrasos, aumento de la calidad, reducción de costes y, en definitiva, mayores beneficios [Gruat La Forme *et al.*, 2007]. De forma general, se puede resumir que con una mayor colaboración entre los miembros de una CdS se consigue mejorar el indicador de rendimiento clave, la rentabilidad de la CdS [Angerhofer y Angelides, 2006].

En la práctica, desde mediados de la década de 1990, la colaboración ha sido entendida y defendida por consultores y académicos en métodos concretos de aplicación a la SCM, como: la gestión de inventarios por parte del proveedor (*Vendor Managed Inventory, VMI*), la Planificación, Previsión y Reposición Colaborativa (*Collaborative Planning Forecasting and Replenishment, CPFR*) y and la Reposición Continua (*Continuous Replenishment, CR*) [Naslund y Williamson, 2010]. Las empresas de hoy en día prefieren relaciones estrechas sólo con algunos socios concretos de sus CdS y la colaboración en general se limita al intercambio de pedidos y a la planificación operativa [Kemppainen y Vepsäläinen, 2007], siendo muy raro encontrar colaboración más allá del primer nivel de relación en la CdS o aguas abajo [Fawcett y Magnan, 2002].

La colaboración entre los participantes deriva en la creación de una red que puede ser:

- Coordinada, en la que el concepto de red queda extendido alineando las actividades de las diversas organizaciones para poder lograr un beneficio mutuo.
- Cooperativa, que extiende la visión coordinada con unos objetivos compatibles que faciliten a cada entidad trabajar de manera aislada pero pudiendo lograr una meta compartida.
- Colaborativa, que extiende la visión de cooperativa con responsabilidades y objetivos conjuntos como una especie de trabajo conjunto que debe ser desarrollado por todos conjuntamente [Jardim-Goncalves *et al.*, 2013].

Una red colaborativa puede materializarse, en función de sus objetivos, en diversas formas, que han sido bautizadas por la comunidad como CdS, organización virtual, empresa virtual, empresa extendida, ecosistema de negocio, cluster industrial, distrito industrial, laboratorio virtual colaborativo, etc como se muestra en la taxonomía de redes colaborativas propuesta por Camarinha y Afsarmanesh [Camarinha-Matos y Afsarmanesh, 2008].

En este trabajo de tesis entenderemos la colaboración en la CdS como un fin deseado, que buscaremos a partir de la definición de los procesos de negocio de los participantes de la CdS con un enfoque inter organizacional, sin profundizar en los niveles de colaboración ni en los aspectos técnicos específicos para conseguir una colaboración operativa.

3. Integración en la CdS

Otro de los términos habituales es el de integración de la CdS, algo que en ocasiones puede verse como una forma de colaboración pero que en otras va más allá. Por ello, se hace necesario de nuevo aclarar los términos y adopciones que ha tenido el término en la literatura, su tipología, beneficios, adopción en la práctica en la industria y cómo entenderemos este término en nuestro trabajo.

Aunque la integración en la CdS no ha sido definida formalmente, existe un conjunto de autores que han intentado acotar su significado en diversos trabajos. Para Lambert et al. [Lambert *et al.*, 1998] el objetivo de la integración de la CdS es mejorar la eficiencia y eficacia del proceso total a través de los miembros de la CdS. Muchos autores ponen énfasis en la importancia que tiene la integración tanto a nivel operativo como estratégico para las CdS [Frohlich y Westbrook, 2001] [Zailani y Rajagopal, 2005], en la que los socios de la CdS se unen para compartir recursos, beneficios y riesgos [Ajmera y Cook, 2009]. En ocasiones la integración se interpreta como una colaboración de alto nivel, donde las partes involucradas actúan como una sola entidad dentro de una empresa extendida [Wen *et al.*, 2007], mientras que en otras la integración se concibe con una perspectiva más amplia y de más largo plazo frente a la colaboración [Newman *et al.*, 2009].

Asociado a esta falta de definición encontramos que existen varios tipos de integración en la CdS. Frohlich y Westbrook presentan dos tipos [Frohlich y Westbrook, 2001]:

- La integración y coordinación del flujo físico de avance de los productos entre proveedor, fabricante y el cliente.
- La integración hacia atrás y la coordinación de la información que fluye aguas arriba en la cadena de suministro.

Fabbe-Costes y Jahre [Fabbe-Costes *et al.*, 2008] identifican cinco ámbitos de integración dependiendo de la naturaleza y el número de empresas incluidas:

1. Diádica y limitada aguas abajo, refiriéndose a la integración entre una empresa y sus clientes.
2. Diádica y limitada aguas arriba, para la integración con sus proveedores.
3. Diádica limitada, para hacer referencia a la integración por separado tanto con clientes como con proveedores.
4. Triádica limitada, en la que se extiende la integración entre el cliente y el proveedor.
5. Extendida: en la que más de tres niveles de participantes de la CdS se integran, por ejemplo, clientes de mi cliente o proveedores de mi proveedor

Cousins y Menguc [Cousins y Menguc, 2006] presentan dos tipos diferentes de integración: la que se manifiesta dentro de una organización y la integración externa, que se centra en las fronteras y relaciones con el resto de compañías de la cadena. En el nivel básico, se consigue una integración de la gestión de los procesos de una organización, de forma que las diferentes áreas funcionales actúan como parte de un proceso integrado y coordinado en lugar de actuar como silos funcionales [Morash y Clinton, 1998]. El segundo nivel, hace referencia a la colaboración entre las diversas organizaciones que forman parte de la CdS [Bowersox, 1990], en la que se llevan a cabo relaciones interactivas y a largo plazo con clientes y proveedores. Se podría argumentar que al igual que la colaboración, la integración también existe en diferentes niveles jerárquicos. En el contexto de la CdS, la integración externa se percibe como un nivel más avanzado en comparación con la integración interna. Newman et al. [Newman *et al.*, 2009] sostienen que la integración externa no tendría éxito sin una exitosa integración a nivel interno en una organización, al igual que Stevens [Stevens, 1990], que defiende que para poder lograr una integración inter organizacional con clientes y proveedores la integración de los procesos dentro de la organización es un prerequisite.

Uno de los beneficios derivados de la integración es la capacidad de la red para diseñar productos más rápido, con mayor calidad y menores costes en comparación con una sola empresa [Ajmera y Cook, 2009]. Ragatz et al. [Ragatz *et al.*, 2002] enumeraron una serie de beneficios potenciales de la integración de la cadena de suministro, entre los que destacan:

- El intercambio de conocimientos, nuevas ideas y tecnologías entre los socios.
- Puede ayudar a identificar problemas y soluciones de forma anticipada.
- Facilita la contratación externa.
- Reduce la complejidad interna de los procesos.
- Puede mejorar la comunicación y el intercambio de información entre las compañías.
- En general, reduce el retrabajo y el coste total de la CdS.

A pesar del elevado número de estudios que resaltan la importancia de la integración en la CdS, no hay evidencias empíricas de su implantación y de sus beneficios, en parte porque no existe en la literatura un método para evaluar la eficiencia de la integración en la práctica [Janvier-James, 2012]. También se hace preciso analizar con mayor detalle los beneficios obtenidos por aquellas organizaciones que de algún modo han desarrollado mecanismos de integración en sus CdS y de los resultados obtenidos en sus indicadores de eficiencia [Panayides, 2006].

En este trabajo de tesis entenderemos la integración en la CdS, al igual que la colaboración, como un objetivo a alcanzar a partir de la definición inter organizacional de los procesos de negocio de los participantes en la CdS, sin profundizar en aspectos concretos o técnicas para que esa integración se haga efectiva.

1.1.3. Modelos y marcos de referencia en la gestión de la cadena de suministro

Un modelo de referencia es una representación abstracta genérica para facilitar la comprensión de las entidades y relaciones entre las entidades de un dominio, así como para creación de otros modelos específicos de casos particulares de ese dominio [Camarinha-Matos y Afsarmanesh, 2008]. Los modelos de referencia crean marcos conceptuales. Su objetivo es incrementar la eficiencia del modelado, aumentar la calidad de los modelos desarrollados y mejorar la reusabilidad de los modelos. Su uso simplifica y agiliza el proceso de modelado y facilita la identificación de estructuras comunes, mejorando la transparencia y la creando estándares para poder establecer comparaciones [Rabe *et al.*, 2006].

En la literatura existen varios modelos y marcos de referencia existentes en el ámbito de la SCM, cuyos principales quedan resumidos a continuación, tomando como base el trabajo realizado por Lambert *et al.* en 2005 [Lambert *et al.*, 2005] y la actualización llevada a cabo por Naslund y Williamson en 2010 [Naslund y Williamson, 2010], en la que se detallan los cuatro principales, cuyas principales características serán descritas a continuación:

- Supply Chain Operations Reference Model (SCOR).
- El modelo GSCF / SCMi.
- El modelo de Planificación, Previsión y Reposición Colaborativa (CPFR).
- El modelo Mentzer.

El modelo SCOR fue desarrollado en el año 1996 por parte del SCC y es en la actualidad el modelo más citado en la literatura [Lockamy III y McCormack, 2004], ofreciendo un único marco en el que se unen procesos de negocio, métricas, buenas prácticas y tecnología dentro de una estructura unificada para dar soporte a la comunicación entre los participantes de la cadena y para mejorar la eficiencia de la SCM. Este modelo es usado para identificar, medir y reorganizar y mejorar los procesos de la CdS a través de un proceso cíclico que incluye:

- La configuración de la SC.
- La medición de la eficiencia de la CdS y su comparación con los objetivos tanto internos como externos.
- La modificación de los procesos y prácticas para alinearlos a los objetivos pendientes de conseguir.

A través de este ciclo SCOR persigue la integración de conceptos bien conocidos como la reingeniería de procesos, el benchmarking y la medición de los procesos dentro de un marco multifuncional [Huan *et al.*, 2004]. Como resultado, organizaciones de muy diversos ámbitos y dominios pueden ser unidas a la hora de describir su cadena de valor [SCC, 2014].

El segundo modelo más citado en la literatura es el desarrollado por el GSCF [Lambert *et al.*, 1998] ahora bajo el paraguas del SCMi. Este modelo se configura en base a ocho procesos clave que permiten llevar a cabo la SCM [Lambert, 2008], [Supply Chain Management Institute (Douglas M. Lambert), 2008]. Cada uno de los procesos se desarrolla a nivel multifuncional con el objetivo de cortar los silos funcionales independientes de cada organización, como son marketing, investigación y desarrollo, finanzas, producción, compras y logística [Lambert *et al.*, 2005]. El marco pone énfasis en la importancia de utilizar un proceso integrador, en el que todas las funciones que intervienen en un producto o están implicadas en el servicio de distribución deben trabajar conjuntamente.

Otro modelo, o más bien una herramienta conceptual, es la Planificación, Previsión y Reposición Colaborativa (Collaborative Planning, Forecasting and Replenishment, CPFR). CPFR se describe como un formato basado en la web creado para coordinar diversas actividades entre los socios comerciales de una CdS, como producción, planificación de compras, previsión de la demanda o la reposición de inventarios. CPFR es el tercer método más utilizado para mejora de la colaboración de la cadena de suministro [Attaran y Attaran, 2007]. El objetivo de CPFR es intercambiar una concreta información interna en un servidor web compartido con el fin de proporcionar una visión más real y a más largo plazo de la demanda dentro de toda la cadena [Fliedner, 2003]. Con esta información extendida se puede obtener una mejor planificación en la CdS, lo que conlleva mejoras como: aumento de ventas, reducción de inventarios y aumento del servicio a los clientes, tanto para los minoristas como para los fabricantes [Cassivi, 2006]. Comparando CPFR con los modelos SCOR y GSCF podemos destacar que CPFR está más centrado en las Tecnologías de la Información y que CPFR no requiere de una masa crítica de usuarios para funcionar, pero permite a una empresa mejorar su rendimiento al tener una colaboración con un miembro de su cadena. Este segundo factor, así como el apoyo brindado desde la asociación VICS (Voluntary InterIndustry Commerce Standards Association), ha facilitado la adopción en la práctica de CPFR en la industria.

En el modelo de Mentzer la CdS se presenta como un oleoducto, que ilustra los flujos de la cadena de suministro, la coordinación inter funcional de las funciones de negocio tradicionales, y la coordinación inter empresarial entre los socios de la cadena, desde los proveedores del proveedor hasta los clientes del cliente para, en última instancia, ofrecer valor y satisfacción para el consumidor. El modelo reconoce en la satisfacción del cliente el factor necesario para poder lograr una ventaja competitiva y una rentabilidad tanto a nivel individual de cada organización

como para la CdS en su conjunto [Mentzer *et al.*, 2001]. A diferencia de los otros modelos presentados que contienen un nivel de detalle a nivel de los procesos para su adopción, el modelo de Mentzer, al estar enfocado en la coordinación inter funcional entre una empresa y las relaciones con el resto de socios en la CdS, los procesos que necesitan ser implementados no están descritos.

A la vista de los últimos marcos comparativos [Naslund y Williamson, 2010], podemos concluir que existen numerosos modelos y marcos de referencia que persiguen estandarizar la forma en la que se definan o se modelen los procesos de negocio en la CdS. Sin embargo, no existe ninguno que haya sido adoptado por la industria como un estándar, quedándose su aplicación en la práctica como guías de buenas prácticas. En este sentido, podemos destacar que hay dos enfoques diferenciados, los que intentan definir los procesos que deberían existir junto con su alcance predeterminado y aquellos que se centran en establecer el marco general de los mismos pero sin entrar en qué procesos deben ser implementados ni cómo llevarlos a cabo.

De cara a nuestro trabajo de tesis entendemos que es posible que una organización no adopte ninguno de estos modelos, o incluso que al participar en varias CdS requiera de ser de alguna forma compatible con varios de ellos, por lo que nuestra solución no será específica para ninguno de estos marcos si bien deberá permitir mejorar la flexibilidad y adaptabilidad de los modelos de procesos de negocio que sean compatibles con varios de estos marcos de referencia.

1.1.4. Estado actual en la gestión de la cadena de suministro

Tras el análisis de las diversas definiciones, marcos y terminología adoptada en la literatura sobre SCM, parece que hay opiniones contradictorias y existe una cierta confusión, tanto en el ámbito académico como en la industria [Stock y Boyer, 2009]. Por ejemplo, mientras que la mayoría de los científicos coinciden en que SCM incluye la integración y la colaboración entre la CdS, todavía no existe una definición comúnmente aceptada, así como un marco que establezca los diversos conceptos sobre cómo debería definirse SCM [Naslund y Williamson, 2010]. Naslund *et al.* resumieron los principales aspectos abiertos en el estado actual en SCM, que presentamos a continuación:

- **Relación uno a uno, no CdS:** Tanto a nivel académico como en la industria la SCM se centra en la totalidad de la cadena o red, desde las materias primas hasta el consumidor final. Sin embargo, en cuanto los estudios se concretan en ciertos problemas en la CdS, el foco se establece en el proceso interno o como mucho en relaciones diádicas. De esta forma, la investigación y la práctica en SCM no parecen extenderse más allá de estas relaciones uno a uno, perdiendo ese carácter global de la CdS.
- **La gestión interna de los procesos, paso previo para SCM:** Para lograr desarrollar la SCM es necesario gestionar los procesos con un enfoque inter funcional. Algunos modelos como SCOR, GSCF y CPFR también promueven la colaboración inter organizacional. Los diversos modelos asumen que para lograr una gestión de procesos inter organizacional es preciso que cada participante de la CdS gestione sus procesos de forma interna.

- **Falta de indicadores clave y dificultad para realizar comparativas entre CdS:** Tan sólo el modelo SCOR incorpora un conjunto de valores establecidos a la hora de evaluar la eficiencia a nivel de los procesos, con un conjunto de datos que pueden servir de referencia para comparar la eficiencia entre diversas CdS. Esto es una de las principales fortalezas de este modelo, si bien dado que las compañías difieren en objetivos, tamaños, dominios de aplicación y procesos, se hace muy difícil establecer comparativas sin disponer de procesos estandarizados.
- **Falta de procesos estándares:** Algunos autores han defendido la necesidad de disponer de un conjunto de procesos de negocio estándares para disponer de una terminología común en el ámbito de la SCM, si bien es un área de trabajo en desarrollo con relativos pocos estudios.
- **La puesta en la práctica de las relaciones entre empresas:** La mayoría de los trabajos resaltan la importancia de establecer relaciones entre empresas para llevar a cabo una colaboración, lo que requiere de generar confianza e intercambiar información. Si bien a nivel teórico está claro que las organizaciones deberían compartir datos con sus proveedores y clientes, en la práctica, esto es visto como un riesgo, ya sea por pérdida de control en las decisiones o en los ámbitos de la seguridad y la privacidad.

En resumen, Naslund et al. [Naslund y Williamson, 2010] destacaron que:

- Es preciso definir claramente SCM así como los conceptos de colaboración e integración asociados a la CdS.
- Es necesario disponer de estudios empíricos que confirmen los beneficios de la SCM en la práctica.
- Las organizaciones tienen muchas dificultades para pasar de la teoría a la práctica implantando de forma exitosa la SCM.
- El mayor reto es ayudar a **diseñar, desarrollar y gestionar procesos de forma inter organizacional** cuando las compañías todavía están en la lucha de gestionar sus procesos de forma interna.

En nuestro trabajo, nos centraremos en ese mayor reto para ser capaces de modelar procesos inter organizacionales incorporando flexibilidad y adaptación. Para entender estas características, vamos a definir los términos en la siguiente sección.

1.1.5. Dinamismo, flexibilidad y adaptación en los procesos

Las trabajos pueden clasificarse de tareas completamente automáticas, rutinarias, regularmente predefinidas o actividades impredecibles y basadas en el conocimiento de los participantes [Breitenmoser y Keller, 2015]. Como puede verse en la figura 1.2, los procesos que incluyen tareas repetitivas pueden ser modelados antes de que se lleven a cabo. Los participantes serán ejecutores de las actividades pero no responsables de la mejora o evolución de los procesos. En el lado derecho, vemos que en aquellos entornos en los que las tareas no pueden ser completamente predefinidas, los participantes son elementos clave no sólo en la ejecución sino en la propia definición de las actividades, por cuanto la organización del proceso se lleva a cabo en tiempo de ejecución atendiendo al conocimiento de las personas.

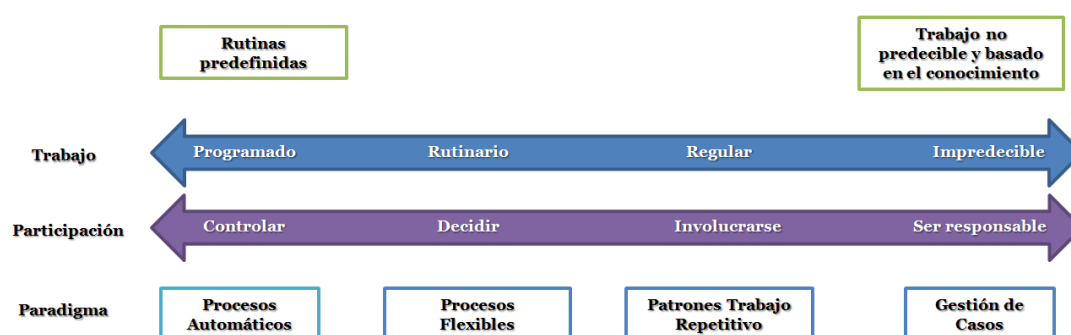


Figura 1.2: Clasificación de trabajos y participación de las personas, adaptado de [Breitenmoser y Keller, 2015]

Debido a los entornos dinámicos y heterogéneos que caracterizan las CdS, es difícil crear un modelo de proceso inter organizacional que describa la colaboración que pueda ser ejecutado desde un inicio hasta el final. La correcta definición y ejecución de este tipo de modelos de procesos requieren métodos y técnicas para, de forma sencilla y eficiente, ser capaces de modificar los modelos o instancias de los mismos, en ocasiones en el propio tiempo de ejecución. Por ello surgen tres conceptos, asociados al dinamismo, adaptación y flexibilidad de los procesos, que pasamos a definir a continuación [Sadiq *et al.*, 2005].

El *dinamismo de un proceso* hace referencia a la modificación de un modelo de proceso. El cambio puede ser pequeño, derivado de alguna corrección o mejora, o drástico, resultado de una alteración significativa de todo el modelo. Existen cuatro principales causas para este dinamismo: una nueva tecnología, nuevos métodos de gestión, nuevas políticas en la organización y nueva legislación. En la actualidad el dinamismo de los procesos está soportado por los sistemas de gestión de workflows y los interpretes y motores de ejecución de procesos de negocio que dan soporte a la ejecución simultánea de diversas versiones o incluso instancias de un mismo modelo de proceso.

La *adaptación de un proceso* implica ajustar un modelo de proceso en una instancia activa para dar soporte a circunstancias excepcionales que puede o no haber sido previstas antes del comienzo de la ejecución de la instancia. La adaptación hace referencia a una o un conjunto de instancias y no implica un cambio permanente en el modelo de proceso. Esto nos puede derivar la pregunta: si un conjunto de situaciones excepcionales pueden ser previstas de antemano, ¿por qué no incluirlas directamente en el modelo del proceso?. La respuesta sería que incluir todas estas circunstancias en el modelo incrementarían significativamente la complejidad del mismo, tanto para su interpretación como para su ejecución. Por ello, para facilitar su adopción en la práctica, es mejor tener un modelo sencillo que incluya las rutas más habituales, crear todas las instancias a partir de dicho modelo, y en el caso de que alguna de estas circunstancias tengan efecto, adaptar el modelo a una instancia en particular.

La *flexibilidad de un proceso* hace referencia a la falta de una especificación completa del modelo de proceso, por lo que el modelo se construye de forma individual y por separado para cada instancia activa del mismo en tiempo de ejecución. El modelo se genera desde actividades individuales predefinidas o desde conjuntos de las mismas, denominadas fragmentos de procesos, que se seleccionan de forma continua de acuerdo con las circunstancias que tienen lugar en la instancia del modelo. Flexibilidad hace referencia a los cambios que un proceso individual puede sufrir atendiendo a unas nuevas circunstancias. En ocasiones se confunde con la variabilidad, si bien este término tiene relación con las diversas diferencias que pueden existir entre un conjunto de procesos [La Rosa *et al.*, 2013].

En este trabajo de tesis cubriremos las CdS en las que se den estas características anteriores, de forma que los modelos de procesos puedan evolucionar de forma dinámica, puedan tener la flexibilidad de participar de forma simultánea en diversas CdS y se adapten a condiciones no previstas en su diseño que tendrán que ser decididas durante su ejecución.

1.2. Los procesos de negocio en la cadena de valor empresarial

Para profundizar en el contexto en el que desarrollamos nuestro trabajo de tesis, en esta sección presentamos los procesos de negocio en el dominio de la CdS y los procesos de negocio colaborativos.

1.2.1. Los procesos de negocio en la cadena de suministro

Las organizaciones se basan en procesos para generar valor al cliente final. El concepto de *proceso* surge en la Revolución Industrial del siglo XVIII y se liga a lo que hoy conocemos como proceso productivo o manufacturero, a veces también generalizado como proceso industrial, esto es, el conjunto de pasos y acciones que se deben llevar a cabo para transformar las materias primas en el producto acabado. La International Organization for Standardization (ISO)⁵ proporciona una definición genérica de proceso, estableciendo que *un proceso consiste en el uso de los recursos para transformar entradas en salidas debido a que algún tipo de trabajo, actividad o función se ha llevado a cabo*.

El concepto de organizar las actividades de una compañía como procesos de negocio fue introducido a finales de 1980 [Davenport *et al.*, 1988] [Hammer y Mangurian, 1990] y llegó a ser popular a comienzos de 1990 tras la publicación de los libros de Hammer y Champy [Hammer y Champy, 1993] y de Davenport [Davenport, 2013a]. Un proceso de negocio es un conjunto estructurado de actividades que derivan en unos resultados enfocados a los clientes [Davenport y Beers, 1995]. Inicialmente los procesos de negocio fueron vistos como el medio para integrar las funciones corporativas dentro de la empresa. En la actualidad, los procesos de negocio son usados para estructurar las actividades entre los diversos miembros de una CdS [Lambert *et al.*, 2005]. El desarrollo de procesos de negocio puede darse en dos niveles: a nivel interno dentro de una organización buscando mejorar la eficiencia y eficacia de las operaciones, y a nivel de la CdS para estructurar las relaciones existentes entre sus miembros. Seguidamente pasamos a exponer las características de cada uno de estos niveles.

A nivel interno, en ocasiones denominada la vista transaccional de la gestión de procesos de negocio, tiene sus raíces en los avances en las tecnologías de la información [Hammer y Mangurian, 1990] y hace referencia a los flujos de trabajo dentro de las organizaciones [Davenport y Short, 1990] tomando como base los principios de Taylor para mejorar la eficiencia en las operaciones de fabricación [Taylor, 1914]. La combinación de procesos de negocio, formando flujos de trabajo, integrados con las tecnologías de la información conducen a una reducción de costes, plazos, aumento de la calidad de los productos permitiendo a las organizaciones disponer de un mayor control sobre su producción [Davenport y Short, 1990]. El objetivo no está en la automatización de los procesos de negocio definidos, sino en el rediseño del modelo para mejorar la satisfacción de los clientes a través de transacciones más eficientes y precisas [Hammer y Mangurian, 1990].

A nivel externo la gestión de procesos de negocio se centra en las relaciones entre las organi-

⁵<http://www.iso.org>

zaciones en la CdS, tomando como origen el concepto de marketing relacional introducido en la década de 1990, cuyo objetivo era establecer, mantener, mejorar y obtener beneficios de las relaciones con clientes y otros socios [Grönroos, 1994]. De esta forma, el desarrollo y mantenimiento de la relación entre partes de una CdS va más allá del cumplimiento de una o de un conjunto de transacciones, con la base de que conseguir la repetición del negocio con un cliente es más rentable que obtener un nuevo cliente [Kotler, 2011]. Las relaciones son más habituales del lado del cliente para lograr la comercialización de los productos, pero el desarrollo y mantenimiento de las relaciones con los proveedores clave de la CdS es igualmente necesario para mantener la sostenibilidad de la cadena a largo plazo. Dicho de otro modo, el éxito empresarial en una CdS se basa en gestionar adecuadamente las relaciones tanto con los clientes como con los proveedores, implicando a personas, organizaciones y procesos [Webster Jr, 1992].

Algunos autores han sugerido la implementación de procesos de negocio en el contexto de la SCM, sin que hasta la fecha exista un estándar adoptado por la industria de cuáles deberían ser esos procesos. Tener procesos de negocio estándar daría la posibilidad a las organizaciones de disponer de un lenguaje común y facilitar la integración de los mismos a lo largo de toda la CdS [Lambert *et al.*, 2005].

Un proceso de negocio de aplicación en la CdS (*Supply Chain Process, SCP*) es un tipo especial de proceso de negocio que relaciona la producción, la logística y la información. Los SPCs tienen un conjunto de características que los hacen diferentes del resto de procesos de negocio:

- Abarcan generalmente diferentes departamentos y son llevados a cabo por diferentes organizaciones independientes.
- Tienen su propia semántica (ej. métricas de rendimiento).
- son utilizados en un gran abanico de operaciones dentro de la organización.

Por ello, se requieren interfaces unificados que faciliten el intercambio de complejas entidades de información que favorezcan la colaboración entre los diversos participantes de la CdS [Xiaodong *et al.*, 2009].

Por todo ello es necesario gestionar los procesos con un enfoque inter organizacional dentro de las CdS, con el objeto de describir, analizar y optimizar las operaciones y transacciones a lo largo de toda la cadena. Si bien todavía existe la asunción de que las organizaciones deben gestionar sus procesos internos en primer lugar, para posteriormente progresar hasta alcanzar esa visión inter organizacional [Naslund y Williamson, 2010]. Una vez evidenciada la necesidad de gestionar procesos con un enfoque inter organizacional para su aplicación en la CdS, en la siguiente sección presentaremos en detalle este tipo de procesos.

1.2.2. Los procesos de negocio colaborativos entre organizaciones

Los procesos de negocio colaborativos (*Collaborative Business Process, CBP*) no han tenido en la literatura una definición consistente, si bien los diversos autores identifican los siguientes criterios para su clasificación [Ziemann, 2010]:

- Número y tipología de organizaciones: en función del número de participantes (no es lo mismo si hay 2 ó 3 entidades que si hablamos de cientos) y el rol que juegan, ya que podemos tener colaboraciones en las que una entidad hace de mediador entre todos hasta entornos en los que cada una establece las comunicaciones directamente con el resto.
- Acoplamiento y cercanía entre las organizaciones: en los entornos en los que las organizaciones se conocen bien y existe confianza mutua se establecen colaboraciones más estrechas que si no se dan estas circunstancias. De esta forma la duración de la colaboración, el nivel de confianza entre los miembros y su dispersidad geográfica son factores que deben abordarse a la hora de implementar un proceso colaborativo.
- Distribución de la coordinación: si hay una entidad que coordina todo el proceso o bien se realiza de forma distribuida.
- Distribución del poder: si una entidad tiene capacidad de mandar sobre el resto (como por ejemplo si hablamos de una gran empresa y su cadena de proveedores) o bien es una red en la que todas las entidades se encuentran al mismo nivel.

Este tipo de procesos, denominados también procesos de negocio inter organizacionales [Bouchbout y Alimazighi, 2011], se caracterizan fundamentalmente porque deben ser llevados a cabo por diferentes organizaciones que comparten un mismo objetivo, y tienen cuatro grandes características:

- La gobernanza del proceso global es descentralizada, no existe una entidad que diseñe, implemente, ejecute y monitorice el proceso de extremo a extremo.
- Debe mantenerse la privacidad de la información individual.
- Debe preservarse la autonomía entre cada uno de los participantes de forma que cada organización diseña, implementa, ejecuta y monitoriza sus procesos internos, manteniendo sus compromisos con el resto.
- La conexión entre las diversas organizaciones se realiza a través de interacciones entre sus procesos.

A la vista de estos criterios podemos definir un proceso colaborativo como aquel en el que sus actividades se ejecutan entre dos o más organizaciones autónomas [Ziemann, 2010]. En esta definición debemos destacar que la colaboración no sólo hace referencia a la interacción entre los diversos procesos de las organizaciones sino entre las personas que los llevan a cabo, y que las entidades participantes actúan como organizaciones autónomas, preservando su información privada y compartiendo exclusivamente aquella necesaria para realizar la colaboración.

En los procesos entre organizaciones, no siempre es posible tener una estructura lógica altamente repetible y completamente conocida a priori, antes de que el proceso se desarrolle y se ejecute. En este sentido han surgido los conocidos como procesos intensivos en conocimiento (*Knowledge-Intensive Processes, KIP*), caracterizados por su alto dinamismo y en los que la lógica de actuación no puede preestablecerse a priori en su totalidad, tomándose la decisión de qué actividades llevar a cabo y en qué orden por alguna persona en tiempo de ejecución [Reichert, 2012]. En este tipo de procesos se busca una relación débilmente acoplada entre los mismos y en lugar de basarse en las actividades la secuencia de actuaciones está dirigida por los datos.

La CdS es un entorno en el que es difícil aplicar cambios en la realidad sin haber analizado las posibles consecuencias, para lo que se hace preciso disponer de técnicas que faciliten su comprensión, análisis y optimización antes de su implementación. Es en este punto donde la generación de modelos de procesos inter organizacionales cobra sentido, para lo cual es necesario disponer de lenguajes que nos permitan generar esos modelos. En la siguiente sección profundizamos en las alternativas existentes.

1.3. Tecnologías de soporte a la gestión de procesos de negocio

En esta sección vamos a poner en contexto las principales tecnologías que vamos a hacer uso durante el desarrollo de esta tesis, como son los lenguajes de modelado de procesos de negocio, la ingeniería del software dirigida por modelos y la computación basada en servicios. Finalizaremos con un resumen de las tendencias en la aplicación de estas tecnologías al dominio de la gestión de procesos de negocio.

1.3.1. Modelado de procesos de negocio

La idea básica de modelar un proceso conlleva la creación de una representación formal del proceso que puede ser sometida a un análisis [Damelio, 2011]. El primer paso es el mapeo de los elementos de proceso reales dentro del modelo, lo que se conoce como la parte estática del modelado, para posteriormente analizar su comportamiento dinámico, en el que se incluyen la simulación de escenarios, el análisis de alternativas o la optimización de tareas o variables [Pawlewski, 2014]. Para llevar a cabo esta tarea se hace uso de lenguajes de modelado.

El modelado de los procesos de negocio tiene cuatro principales objetivos [Kosanke, 2009]:

- Adquirir conocimiento explícito sobre los procesos operativos de la organización.
- Explotar este conocimiento para mejorar dicha operativa a través de la reingeniería de los procesos.
- Dar soporte a la toma de decisiones en la organización.
- Facilitar la interoperabilidad entre los procesos de la compañía.

Los dos últimos objetivos están intrínsecamente ligados a la colaboración inter organizacional, siendo necesario disponer de una interoperabilidad entre los modelos de proceso de negocio que permita unificar en un único modelo los diversos modelos de cada participante para analizar y optimizar nuevos escenarios.

En los últimos años se han desarrollado más de 50 lenguajes para describir procesos [Lucas *et al.*, 2005]. Aunque la estandarización de las propuestas es un tema vigente desde hace más de 10 años, ninguna de las propuestas es hoy aceptada como estándar de facto en la industria [Heidari *et al.*, 2013].

El enfoque tradicional para BPM se centra principalmente en los aspectos del proceso, dejando en un segundo lugar los datos o la información que se genera durante su ejecución [Hull *et al.*, 2013]. Por ejemplo, BPMN asume que los procesos se ejecutan en base a una secuencia de actividades predeterminada y no da un soporte adecuado para controlar procesos en los que las decisiones las toman personas [de Man, 2009]. Sin embargo, los datos pueden tener mucha importancia dentro de un proceso de negocio [Hull *et al.*, 2013], distinguiendo entre:

- Los datos del negocio en el proceso (ej. dirección de entrega de una compra).
- Los datos que representan el estado de la ejecución de una actividad (ej. si el envío se ha realizado).
- Los datos sobre el uso de los recursos asociados para llevar a cabo la actividad (ej. camión reservado para hacer la entrega).
- Los datos que describen las relaciones entre las diversas instancias de un proceso que pueden estar en ejecución en un momento dado (ej. cinco compras en paralelo, siguiendo el mismo proceso de compra, pueden encontrarse en estados diferentes).

Atendiendo a cómo dan cobertura a los datos dentro de los procesos, Hull et. al [Hull *et al.*, 2013] agrupan los lenguajes de modelado de procesos en cuatro categorías:

- Aquellos que ignoran los datos excepto en lo relativo a la sincronización de las actividades (ej. redes de Petri, diagramas de actividad UML, BPMN).
- Los que son conscientes de los datos para especificar la lógica del proceso (ej. BPEL, YAWL).
- Los que guardan los datos de forma persistente, si bien no trazan la relación con el proceso (ej. jBPMN).
- Los basados en artefactos [Vaculin *et al.*, 2011] (*artifact-centric modeling*), que parten del modelado de los datos como un elemento fundamental para gestionar la evolución de un modelo de proceso. El concepto de artefactos del negocio fue introducido en 2003 por Nigam y Caswell [Nigam y Caswell, 2003], como un elemento para capturar los elementos conceptuales clave que fluyen en el proceso, siendo preciso especificar: 1) Un modelo de información (o esquema de datos), para guardar los datos sobre el artefacto conforme se mueve a lo largo del proceso desde su creación, y; 2) El esquema de su ciclo de vida que define cómo y cuándo las actividades podrían actuar sobre el artefacto durante el proceso.

Para dar soporte al modelado centrado en artefactos de negocio, han surgido y se han aplicado diversos enfoques y lenguajes en el ámbito de la gestión de procesos de negocio, cuyas diferencias se presentan en cómo se lleva a cabo la especificación del ciclo de vida de dichos artefactos. De forma resumida podemos destacar:

- Las máquinas de estado finitas [Bhattacharya *et al.*, 2007][Chao *et al.*, 2009] y su variante para cubrir los objetos de negocio [Künzle y Reichert, 2011][Redding *et al.*, 2009].
- El enfoque *Guard-Stage-Milestone (GSM)* [Hull *et al.*, 2010][Damaggio *et al.*, 2013], basado en metas y más declarativo, con utilidad especial para dar soporte a procesos en los que exista colaboración o decisiones tomadas por una persona en base a su conocimiento (*knowledge-worker-driven BPs*) [Hull *et al.*, 2013].
- *EZ-Flow* [Xu *et al.*, 2011], una variación que se centra en el flujo de los datos y persigue la optimización lógica de los artefactos.

Con carácter general, los diversos enfoques de gestión de procesos se centran en las actividades, dejando los datos como elementos persistentes que van dejando huella de las operaciones realizadas. Cómo poder incluir los datos como elementos del modelo de proceso es uno de los nuevos temas de investigación que surgieron de la comunidad de BPM y que han derivado en lo que conocemos como gestión de casos. Por ello, la gestión de casos surge de la comunidad de BPM para dar soporte a [de Man, 2009]:

- Cómo incluir los datos como un elemento clave en el flujo de un proceso.
- Cómo representar procesos cuyo flujo está determinado por las decisiones de un trabajador en base a su conocimiento.
- Cómo gestionar la ejecución de un proceso cuando no es posible determinar a priori toda la secuencia de acciones.

1.3.2. La gestión de casos

La gestión de casos (conocida en la literatura como *case management* o *case handling*) fue introducida por primera vez en la literatura dentro del ámbito de la gestión de procesos por Van der Aalst et al. en 2005 [Van der Aalst *et al.*, 2005], quienes la definieron como un nuevo paradigma para dar soporte a la flexibilidad y al conocimiento intensivo en los procesos de negocio. Propusieron un metamodelo de gestión de casos, mostrado en la figura 1.3.

La gestión de casos da soporte al conocimiento de las personas en aquellos escenarios que requieren un nivel de flexibilidad mayor que los flujos de trabajo convencionales de los procesos de negocio [Marin *et al.*, 2013]. Las características de la gestión de casos son [de Man, 2009]:

- Los datos son el contexto central y compartido que dirige las actividades.
- Su ejecución está basada en eventos, en control de su estado interno y en una fuerte colaboración entre los participantes.
- El factor humano es predominante, ya que toma decisiones conforme se avanza en el caso, atendiendo a las alternativas previamente definidas .

Existen diversos paradigmas para modelar casos, distinguiendo entre los basados en actividades, artefactos y comunicaciones, si bien en la actualidad ninguno de ellos dispone de la capacidad de modelar el factor humano, esto es, el grado de libertad de las personas que intervienen en un caso para decidir qué actividades incluir en el mismo [de Man, 2009].

Podemos diferenciar entre la gestión de casos productivos (*Production Case Management, PCM*) y adaptativos (*Adaptive Case Management, ACM*). PCM da soporte a los procesos que pueden ser completamente definidos en tiempo de diseño. ACM hace referencia al trabajo necesario para dar soporte a un caso de una forma flexible, tomando decisiones en función del entorno y del contexto para seleccionar las alternativas de alguna forma se han predeterminado [Motahari-Nezhad y Swenson, 2013]. Sus principales características se muestran en la figura 1.4.

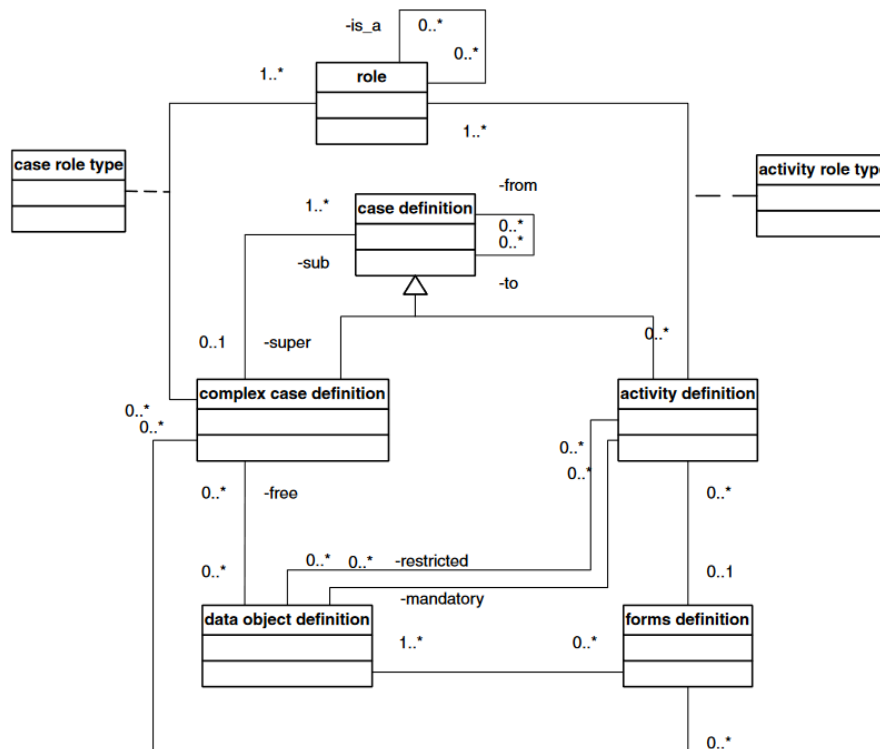


Figura 1.3: Metamodelo de gestión de casos propuesto por Van der Aalst et al. [Van der Aalst et al., 2005]

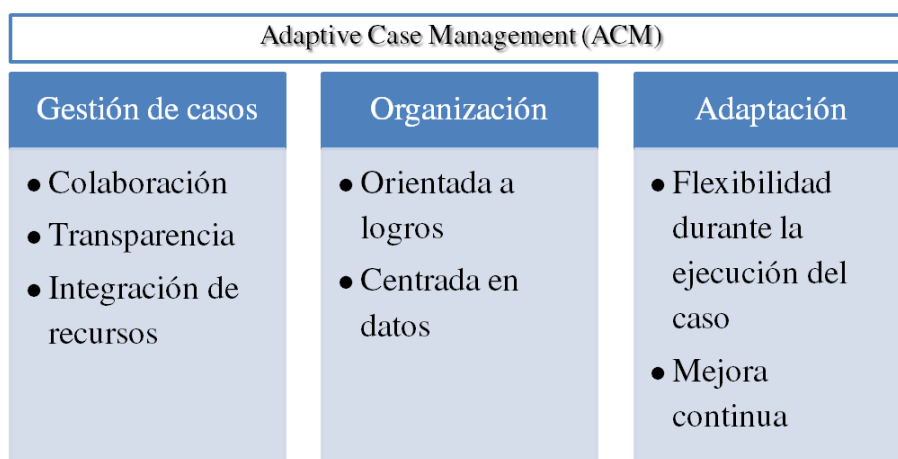


Figura 1.4: Características de los Adaptive Case Management [Herrmann y Kurz, 2011]

Esa habilidad para adaptar procesos en tiempo dinámico de ACM está ganando mucha relevancia en la literatura [Swenson *et al.*, 2010] y posiciona a las organizaciones que lo consigan con capacidades de responder rápidamente ante cambios gestionando mejor su conocimiento y las incertidumbres [Taleb, 2012].

La revisión de la literatura llevada a cabo por Hauder *et al.* [Hauder *et al.*, 2014] concluye que la gestión de casos emerge como un paradigma válido para la gestión de KIPs, siendo ACM el enfoque que más atención está ganando en los campos de la investigación, la estandarización y la industria. Los autores destacan que siendo un campo inmaduro todavía no existe un modelo conceptual o una base teórica unificada para ACM.

Por todo ello consideramos que el paradigma de la gestión de casos, y más concretamente su versión ACM, encajan perfectamente en las necesidades de dinamismo, flexibilidad y adaptación descritas para el modelado de procesos colaborativos. Seguidamente vamos a exponer la notación *Case Management Model and Notation (CMMN)*, la apuesta del OMG por estandarizar el dominio de la gestión de casos, que mantiene y amplía los conceptos introducidos por Van der Aalst *et al.* [Van der Aalst *et al.*, 2005].

En 2008 el OMG comenzó a trabajar en el modelado de casos con estos objetivos de diseño [de Man, 2009]:

- Los datos del caso deben servir como un contexto central y compartido sobre todas las actividades que se llevan a cabo.
- La gestión de casos es altamente colaborativa y basada en eventos por naturaleza.
- El factor humano es predominante. Por ello, las decisiones que toma un trabajador de un caso pueden influir en el proceso conforme evoluciona. Si bien estas decisiones se basan en su experiencia y conocimiento, se debería poder guiar al trabajador en todo lo posible.

En 2009 el OMG [Object Management Group, 2009] lanzó una llamada de propuestas para extender la notación BPMN con el objetivo de soportar el modelado de casos. Como resultado de este trabajo en 2013 surgió una versión beta y en 2014 el OMG [OMG, 2015a] publicó la especificación *Case Management Model and Notation (CMMN)* [OMG, 2014], que define un metamodelo y una notación para modelar gráficamente un caso, así como un formato para el intercambio de modelos de casos entre diversas herramientas.

El contexto de CMMN, un caso tiene dos fases muy diferenciadas:

- La fase de diseño (*design-time phase*), en la que el ingeniero de procesos, desarrollador o analista del negocio realiza el modelo del caso. En él se establecen las tareas que deben realizarse de forma obligatoria, otras que puedan ser opcionales durante su ejecución y finalmente aquellas que, en función de la elección del trabajador del caso podrán ser planificadas de forma dinámica y se modelarán como discrecionales.
- La fase de planificación y ejecución (*run-time planning and execution phase*), en la que los trabajadores del caso (*case workers*) crean y ejecutan un plan, llevando a cabo las tareas

planificadas. Dado que el plan evoluciona de forma continua, e incluso nuevos trabajadores pueden formar parte del caso, es en tiempo de ejecución cuando se pueden volver a planificar tareas (incluir opcionales o discrecionales), dotando de flexibilidad al modelo para dar respuesta a nuevas necesidades que surjan durante la ejecución.

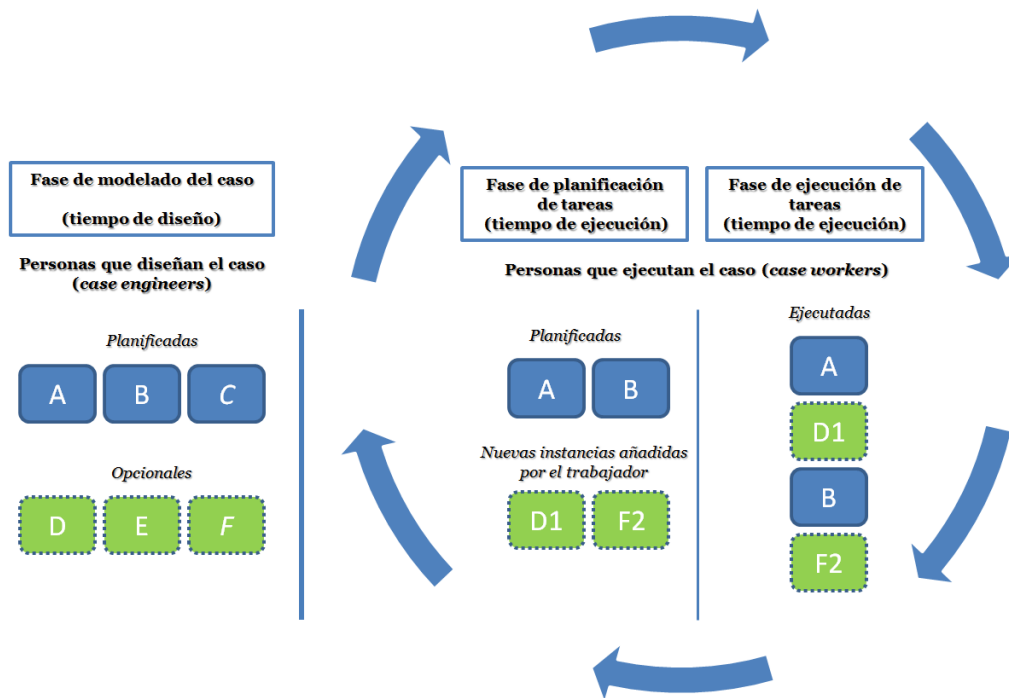


Figura 1.5: Fases de diseño y planificación-ejecución del modelo CMMN

La figura 1.5 resume esta diferenciación de la fase de diseño y fase de planificación y ejecución. En ella vemos que se ha modelado un caso con seis tareas, tres están planificadas (A, B y C) y otras tres son discrecionales a juicio del trabajador del caso (D, E y F). Adicionalmente las tareas en cursiva (C y F) son opcionales. En tiempo de planificación, el trabajador del caso puede añadir instancias de tareas discrecionales (en el ejemplo D1 y F2), que se añaden dinámicamente al caso por lo que en este momento se han ejecutado cuatro tareas (A, D1, B y F2). En esta fase se tiene un ciclo continuo de planificación y ejecución y es lo que permite la evolución del modelo del caso otorgándole la flexibilidad para atender nuevas necesidades.

Los principales elementos del metamodelo de CMMN quedan reflejados en la figure 1.6, elaborada a partir de la especificación oficial [OMG, 2014].

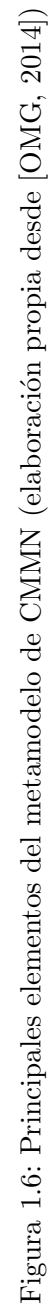


Figura 1.6: Principales elementos del metamodelo de CMMN (elaboración propia desde [OMG, 2014])

Si bien se trata de un estándar adoptado por el OMG, existen un conjunto de elementos que han dificultado su adopción como tal:

- La falta de herramientas que den soporte tanto al modelado como a la ejecución de casos. Los principales fabricantes no han adoptado el estándar y aquellos que lo han hecho, como Camunda [services GmbH, 2016], no dan soporte a todos los elementos.
- La falta de estudios empíricos y casos prácticos. Aunque se han realizado casos piloto para validar su posible adopción práctica, en dominios como el software [Miniet y Navarro, 2014] o la sanidad [Herzberg *et al.*, 2014] [Marin y Brown, 2015], sus conclusiones evidencian la necesidad de adoptar cambios y mejoras al lenguaje.
- La existencia de errores e inconsistencias en el propio modelo, algo que se ha ido recogiendo dentro del OMG y que ha derivado en una *Revision Task Force (RTF)* actualmente en curso.

Por todo ello desde el OMG se ha trabajado en una nueva versión del estándar, la versión 1.1 [OMG, 2016], actualmente disponible en versión beta que pretende:

- Corregir errores descubiertos en la especificación.
- Incrementar la claridad de la especificación.
- Corregir algunos elementos de cara a mejorar su implementación y facilitar su adopción en la práctica como estándar.

Entre los cambios propuestos encontramos una mejora en la representación visual de algunos elementos, nuevos componentes como la *Decision Task* y la capacidad de extensión para fabricantes de herramientas.

1.3.3. La ingeniería del software dirigida por modelos

La ingeniería del software dirigida por modelos (*Model-Driven Engineering, MDE*) es un paradigma metodológico de desarrollo de software que se centra en la creación y explotación de modelos de dominio (es decir, representaciones abstractas de los conocimientos y las actividades que rigen un dominio de aplicación particular) y no en algoritmos [Schmidt, 2006].

Existen muchos acrónimos dentro del dominio de MDE, con el formato MD* (*Model-Driven star*), que representan particularidades dentro del dominio de los procesos basados en modelos. El desarrollo dirigido por modelos (*Model-Driven Development, MDD*), es el paradigma de desarrollo de software que usa modelos como el elemento básico del proceso de desarrollo. En MDD la implementación se genera parcialmente de forma automática. La arquitectura dirigida por modelos [2000] (*Model-Driven Architecture, MDA*), es la visión particular de MDD propuesta por el OMG [OMG, 2015a] basada en los estándares del OMG. MDA puede ser vista como un subconjunto de MDD en el que los lenguajes de modelado y de transformación han sido estandarizados por el OMG. La ingeniería del software dirigida por modelos, en ocasiones también denominado *Model-Driven Software Engineering, MDSE*), es un superconjunto de MDD, ya que va más allá del puro desarrollo de software y engloba otras actividades basadas en modelos como la evolución o la ingeniería inversa. La ingeniería basada por modelos (*Model-Based Engineering, MBE*), es un superconjunto de MDE en el sentido de que los modelos de software no son el elemento fundamental del proceso de desarrollo.

MDE es una metodología que aporta las ventajas del modelado en la ingeniería del software, incorporando: conceptos, notaciones, procesos, reglas y herramientas de soporte [Brambilla *et al.*, 2012a]. Los elementos fundamentales de MDE son los modelos y las transformaciones entre modelos [Cetinkaya y Verbraeck, 2011], que deben ser expresados mediante alguna notación, lo que en MDE se denomina un lenguaje de modelado, y define la sintaxis o notación del modelo así como su semántica o significado. Las transformaciones nos permiten derivar unos modelos a partir de otros, mediante una relación entre dos sintaxis abstractas que se concreta con las relaciones entre los elementos de los correspondientes metamodelos [Thiry y Thirion, 2009].

En MDE todo son modelos (*«Everything is a model»*) [Bézivin, 2005], ya que las propias transformaciones pueden ser vistas como modelos de operaciones sobre otros modelos. Incluso la propia definición de un lenguaje de modelado puede ser visto como un modelo, algo que en MDE se conoce como metamodelado (ej. modelar un modelo o modelar un lenguaje de modelado). Y además se puede aplicar esta técnica de forma recursiva, modelar un metamodelo para crear un meta-metamodelo. El paradigma basado en modelos implica una gran diferencia frente a dibujar o representar un sistema, ya que nos permite: validar sintácticamente, comprobar la validez de un modelo (model-checking), simular un modelo, transformar el modelo en otro o incluso ejecutar un modelo [Brambilla *et al.*, 2012a]. Un metamodelo es por tanto un tipo especial de modelo que especifica un lenguaje de modelado [Mellor, 2004], definiendo la estructura y restricciones para una familia de modelos.

Los lenguajes de modelado son el mecanismo que permiten a los diseñadores especificar los modelos de sus procesos o sistemas. Establecen la forma en la que se define la representación concreta de un modelo conceptual y pueden componerse de representaciones gráficas, textuales o incluso ambas. En cualquier caso, los lenguajes de modelado están formalmente definidos y obligan a los diseñadores a hacer uso de sus sintaxis a la hora de crear los modelos [Brambilla *et al.*, 2012a].

Existen dos grandes grupos de lenguajes de modelado.

- Los lenguajes específicos de dominio (*Domain-Specific Languages, DSLs* o en ocasiones *Domain-Specific Modeling Languages, DSMLs*), que se diseñan de forma específica para cierto dominio, como por ejemplo HTML, Logo, VHDL, Matlab o SQL.
- Los lenguajes de modelado de propósito general (*General-Purpose Modeling Languages, GPMLs* o en ocasiones *General Modeling Languages, GMLs* o incluso *General-Purpose Languages, GPLs*), que pueden ser utilizados para cualquier dominio de aplicación, como por ejemplo UML, las redes de Petri o las máquinas de estado.

El enfoque basado en modelos para representar procesos en un dominio específico presenta una serie de ventajas como [Cetinkaya y Verbraeck, 2011]:

- Tener una forma común de representar procesos.
- Facilitar la compatibilidad con otros formalismos.
- Disponer de herramientas de soporte para la ejecución de los mismos.
- Facilitar la reutilización de modelos.
- Poder estar orientado a crear soluciones específicas de dominio y permite simular modelos como soporte a la toma de decisiones [Mohagheghi *et al.*, 2013].

MDE es un paradigma metodológico que se centra en la creación y explotación de modelos de dominio [Schmidt, 2006]. Pero los modelos no son entidades estáticas aisladas, sino que pueden ser [Brambilla *et al.*, 2012b]:

- Combinados (ej. para homogeneizar diversas versiones de un sistema).
- Alineados (ej. para crear una representación global de un sistema desde distintos puntos de vista).
- Rehechos (ej. para mejorar su estructura interna sin modificar su comportamiento observable desde el exterior).
- Redefinidos (ej. para detallar el modelo con un mayor nivel de abstracción).
- Traducidos (ej. a otros lenguajes o dominios, o como parte de la generación automática de código o a procesos de simulación).

En MDE, las operaciones sobre los modelos se realizan en base a transformaciones [Sendall y Kozaczynski, 2003], pudiendo ser de varios tipos:

- Transformación Modelo a Modelo (*model-to-model*, *M2M*): cuando tanto la entrada como la salida se corresponden con un modelo.
- Transformación Modelo a Texto (*model-to-text*, *M2T*): cuando a partir de un modelo de entrada la salida es una cadena de texto.
- Transformación Texto a Modelo (*text-to-model*, *TM2*): cuando a partir de una cadena de texto la salida es un modelo, siendo la ingeniería inversa su dominio de aplicación más habitual.

La figura 1.7 representa los tipos de transformaciones, en ella podemos ver:

- Una transformación M2M desde el modelo A al modelo B.
- Una transformación M2T desde el modelo A al texto D.
- Una transformación T2M desde el texto E al modelo C.

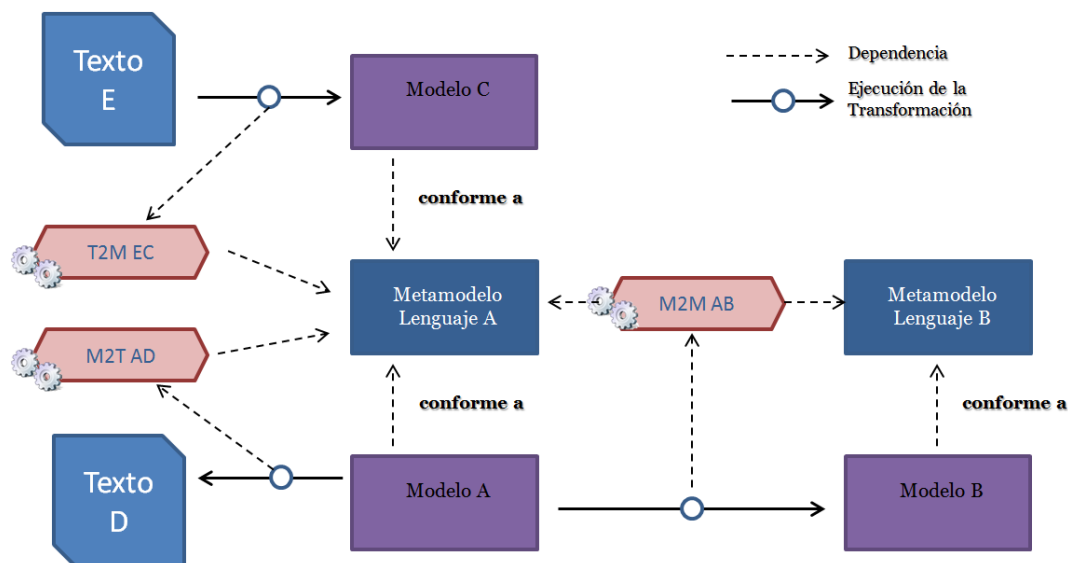


Figura 1.7: Transformaciones M2M y M2T

Una transformación M2M se define como *la generación automática de un modelo destino a partir de un modelo origen de acuerdo a un conjunto de reglas de transformación* [Kleppe et al., 2003]. Una regla de transformación es una descripción de cómo una o mas construcciones de un lenguaje origen puede ser convertida en una o más construcciones en el lenguaje destino.

Una transformación M2M en un sentido general es un programa que toma como entrada uno o varios modelos y genera uno o varios modelos de salida. En este proceso las combinaciones posibles atendiendo al número de entradas y salidas son:

- Transformación uno-a-uno (*one-to-one*): cuando desde un modelo se genera otro modelo, siendo el escenario más habitual.
- Transformación uno-a-muchos (*one-to-many*): cuando desde un modelo se generan varios modelos de salida.
- Transformación muchos-a-uno (*many-to-one*): cuando desde varios modelos de entrada se obtiene un único modelo.
- Transformación muchos-a-muchos (*many-to-many*): cuando desde varios modelos de entrada se generan varios modelos de salida.

Otra posible clasificación se obtiene, no en función del número de entradas y salidas, sino atendiendo al lenguaje de definición de los modelos, teniendo [Mens y Van Gorp, 2006]:

- Transformaciones exógenas: cuando los modelos de entrada y salida se corresponden a distintos lenguajes de modelado.
- Transformaciones endógenas: cuando los modelos de entrada y salida comparten el mismo metamodelo.

La figura 1.8 muestra dos transformaciones M2M, una exógena (M2M AB), que transforma dos modelos que corresponden a lenguajes diferentes y otra endógena (M2M AC), en la que los modelos A y C corresponden al mismo metamodelo.

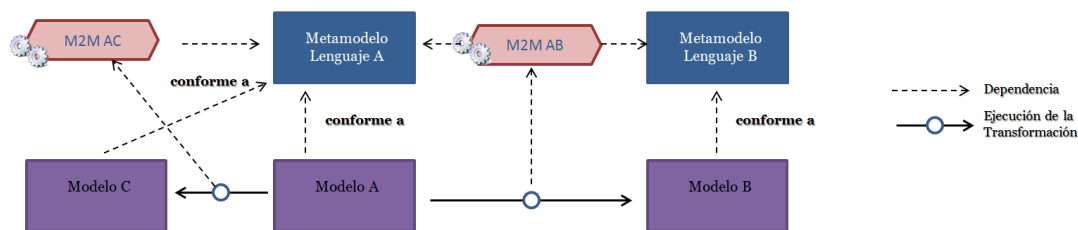


Figura 1.8: Transformaciones M2M endógenas y exógenas

Otra posible clasificación de las transformaciones es en base al nivel de abstracción de las mismas, teniendo:

- Transformaciones horizontales: cuando los modelos de entrada y salida están al mismo nivel de abstracción, por ejemplo ambos a nivel PIM o PSM). Se utilizan en casos de refactorización (sería además endógena) o migración (sería además exógena).
- Transformaciones verticales: cuando los modelos de entrada y salida están a diferentes niveles de abstracción. Un ejemplo típico es el refinamiento [Back y Wright, 2012], en el que la especificación se redefine de forma gradual en implementaciones que añaden detalles concretos [Wirth, 1971].

	Horizontal	Vertical
Endógena	Refactorización	Refinamiento Formal
Exógena	Migración de Lenguaje	Generación de Código

Figura 1.9: Dimensiones ortogonales de las transformaciones M2M con ejemplos

La figura 1.9 muestra las dimensiones ortogonales de las transformaciones M2M, atendiendo a si son endógenas o exógenas, en una dimensión, y verticales u horizontales en la otra.

Como veremos en el capítulo 2 existen un conjunto de propuestas basadas en modelos de aplicación a los procesos de negocio de la CdS, por lo que va a ser un paradigma que creemos es de aplicación y que constituye de hecho nuestra hipótesis principal de investigación. De igual forma, las transformaciones entre modelos serán una influencia tecnológica que tomaremos como base a la hora de abordarlas dentro de la solución propuesta.

1.3.4. La computación basada en servicios

La computación basada en servicios (*Service-Oriented Computing*, SOC) es el paradigma computacional que utiliza los servicios como elemento fundamental para el desarrollo de aplicaciones [Papazoglou, 2003]. Para construir el modelo de los servicios, SOC se apoya en las arquitecturas orientadas a servicio (*Service-Oriented Architectures*, SOA), que es la forma de reorganizar las aplicaciones y las infraestructuras en torno a un conjunto de servicios que interactúan [Papazoglou y Van Den Heuvel, 2007b].

Los servicios, sus descripciones y sus operaciones básicas (publicación, descubrimiento, selección y enlace) que producen o utilizan esas descripciones constituyen las bases fundamentales de SOA. Un servicio es un componente abierto, auto-descriptivo, que ayuda el desarrollo de aplicaciones rápidas, distribuidas y basadas en la composición de servicios. Dado que los servicios pueden ser accesibles bajo diversos protocolos a través de internet, ofrecen una infraestructura de computación distribuida para la integración y colaboración de aplicaciones tanto dentro como entre distintas organizaciones [Papazoglou, 2003].

La visión de SOC es crear un mundo de servicios que cooperan en el que los componentes de las aplicaciones se integran con poco esfuerzo en una red de servicios débilmente acoplados para dar soporte a procesos de negocio flexibles y dinámicos y al desarrollo de aplicaciones ágiles que pueden abarcar varias organizaciones y tecnologías [Leymann, 2005]. Con el paradigma SOC, sistemas legados independientes como los de la gestión de la CdS, pueden ser convertidos en arquitecturas de servicios, facilitando la integración entre las organizaciones [Leymann, 2003].

Las aplicaciones SOA se construyen con la premisa de que todas las organizaciones tienen un diseño de su negocio, que describe cómo el negocio se desarrolla, en base a sus procesos, estructura organizativa, objetivos, influencias del mercado, reglas y políticas internas [Parazoglou, 2006]. La base del diseño del negocio son los procesos de negocio que contribuyen en la definición de cómo la organización actúa y responde ante los clientes, las oportunidades y las amenazas internas y externas [Leymann y Roller, 2000]. Por ello, SOA se apoya en un enfoque de ingeniería del software evolutivo que se basa en parte en los procesos de desarrollo de software anteriores, incluyendo el desarrollo basado en componentes y el modelado de los procesos de negocio [Harmon, 2003].

Al igual que en el dominio de la SCM, en SOA también existen multitud de modelos, marcos y arquitecturas de referencia, sin que ninguno haya sido admitido por la industria como estándar ni existan formas para evaluar su conveniencia o seleccionar el más adecuado para un dominio en particular, estando algunos de ellos orientados a la creación de redes de organizaciones colaborativas [de Oliveira *et al.*, 2010]. A su vez, en los últimos años han surgido un conjunto de especificaciones y estándares en el marco de SOA de la mano de organizaciones como OASIS [OASIS, 2015], el OMG [OMG, 2015a] o el OpenGroup [OpenGroup, 2015]. Entre ellas destacan: el modelo de referencia de SOA creado por OASIS [Standard, 2006], la arquitectura de referencia para los fundamentos SOA creada por OASIS [Standard, 2008], el lenguaje de modelado SOAML propuesto por el OMG [Berre, 2008], la ontología SOA definida por el OpenGroup [Harding, 2008] o el marco de gobernanza SOA del OpenGroup [Framework, 2009]. Si bien entre ellos parece que hay un elevado grado de acuerdo en los conceptos básicos que constituyen SOA, la existencia de tantas propuestas no integradas dificulta la adopción en la práctica por parte de las organizaciones [Kreger y Estefan, 2009].

Típicamente SOA y la gestión de procesos de negocio (*Business Process Management, BPM*) han sido dos disciplinas divergentes [Bajwa *et al.*, 2008]. BPM es una actividad dirigida por el negocio que persigue gestionar los procesos de una organización, en base a tres fases: modelado del proceso, implementación y optimización [Ashton y Kelly, 2006]. SOA es una actividad dirigida por la tecnología que facilita la comunicación de servicios débilmente acoplados y fuertemente interoperables [Bhombal,]. En la actualidad la combinación de BPM con SOA se ha establecido como la mejor propuesta que las organizaciones tienen para conseguir un alineamiento entre sus procesos de negocio y las infraestructuras de soporte y alcanzar la agilidad y flexibilidad necesarias para dar respuesta a los cambios en el negocio [Kamoun, 2007]. BPM permite modelar un proceso de negocio extremo a extremo más allá de las fronteras tradicionales de una organización y sus sistemas. SOA permite crear sistemas a partir de reusable servicios del negocio, por lo que las aplicaciones interoperan sin estar acopladas. Cuando se combinan, las organizaciones deben implementar sus procesos de negocio como servicios y las herramientas de soporte a BPM pasan a ser aplicaciones que componen servicios [Bajwa *et al.*, 2009]. En base a la literatura, resumimos algunas de las diferencias y similitudes entre BPM y SOA en la tabla 1.1.

El modelado de los procesos de negocio puede por tanto contribuir a crear convenciones que ayuden a analizar las cadenas de valor y mejorar sus procesos, dar soporte a la creación de procesos y procedimientos operativos estandarizados y crear un conocimiento compartido de cómo funciona una organización de manera que las implantaciones de los flujos de trabajo pueden ser testeadas antes de su diseño y desarrollo final [Papazoglou *et al.*, 2008]. Juntos, BPM y SOA facilitan la siguiente fase de la evolución de los procesos de negocio para dar soporte a

BPM	SOA	Similitudes
Dirigido por el negocio	Dirigido por la tecnología	Ambos dan soporte a los cambios dinámicos
Enfoque de proceso de arriba a abajo	Enfoque arquitectónico de abajo a arriba	Ambos siguen un proceso iterativo
Reutiliza modelos de proceso	Reutiliza implementaciones de servicios	Ambos apuestan por la reutilización
Orientado a proyectos	Orientado a infraestructuras para la organización	Ambos persiguen un débil acoplamiento
El éxito es medido a través de métricas e indicadores del negocio (<i>Key Performance Indicators, KPIs</i>)	El éxito es medido a través de métricas sobre la arquitectura, como la facilidad de integración o el coste de realizar cambios	Ambos se desarrollan en un entorno distribuido

Tabla 1.1: Diferencias y similitudes entre BPM y SOA

su flexibilidad y adaptación [Bajwa *et al.*, 2008], ya que su combinación presenta las siguientes ventajas:

- Reduce el coste operativo, de desarrollo y de mantenimiento del negocio.
- Acelera la creación y modificación de los procesos.
- Da soporte a incrementar la eficiencia global de una operación.
- Mediante la reutilización se reduce la complejidad.
- Ofrece agilidad, flexibilidad y capacidad de adaptación a las empresas para incluir cambios dinámicos.

En cuanto a los trabajos realizados en el campo de SOA y BPM de forma integrada, las actividades se han concentrado en dos grandes áreas para desarrollar una metodología para la ingeniería orientada a servicios (*Service-Oriented Engineering, SOE*) y en el desarrollo de modelos en tiempo de diseño. Por un lado se han propuesto elementos para crear metodologías para el diseño y desarrollo de servicios, dando principios y guías para especificar, construir y refinar procesos de negocio volátiles que son coreografiados a través de servicios web propios o ajenos a la organización [Ghezzi, 2005] [Papazoglou y Van Den Heuvel, 2007a]. Por otro se han desarrollado modelos en tiempo de diseño utilizando técnicas de análisis de requisitos basadas en metas [Van Lamsweerde, 2000], en las que los objetivos se pueden hacer accesibles en tiempo de ejecución dentro de la infraestructura de descubrimiento de servicios y de esa forma facilitar la selección del servicio a componer que cubra las necesidades [Kaabi *et al.*, 2004] [Penserini *et al.*, 2006].

La organización OASIS [OASIS, 2015] define SOA como *el paradigma para organizar y utilizar las capacidades distribuidas que pueden estar bajo el control de diferentes dominios* [Standard, 2006], poniendo de manifiesto alguna de las características de SOA para facilitar la colaboración entre organizaciones en la consecución de un objetivo común. OASIS define un servicio como *el mecanismo por el cual se alinean necesidades y capacidades* [Standard, 2006]. En un contexto de colaboración entre organizaciones, esta definición de servicio se centra exactamente en cubrir las necesidades de los usuarios con las capacidades de una o de un conjunto de organizaciones que dan respuesta a las mismas [Picard *et al.*, 2014]. Por ello el paradigma de SOA ha sido comparado con los entornos inter organizacionales, en particular con las características que tienen las redes de organizaciones colaborativas (*Collaborative Networked Organizations, CNO*) [Camarinha-Matos y Afsarmanesh, 2008], entendidas como redes compuestas por una variedad de actores autónomos, distribuidos geográficamente y con un entorno heterogéneo a nivel tecnológico, cultural, capital social y objetivos, que llevan a cabo procesos en los que al menos existe una colaboración para cubrir las necesidades de un cliente. Existen por tanto un conjunto de características dentro de SOA que pueden ser de aplicación en un entorno inter organizacional, como se muestra en la tabla 1.2.

La aplicación de SOA a nivel inter organizacional tiene dos características específicas, su alta regulación y su entorno altamente competitivo y dinámico. La regulación deriva en el problema de la flexibilidad y el dinamismo conlleva el problema de la adaptación de los procesos colaborativos. En los últimos años encontramos iniciativas que intentan abordar estos problemas, como los protocolos de servicio y la propuesta CMEAP (Composable Modeling and Execution of Administrative Procedures), que ofrecen un método para especificar, instanciar y adaptar modelos de procesos colaborativos y protocolos de servicios [Picard, 2013] [Paszkiewicz y Cellary, 2012].

Como conclusión hemos visto que el paradigma SOA tiene un conjunto de características que hacen viable su adopción para el modelado de procesos de negocio inter organizacionales, en particular para poder llegar a cubrir las características de flexibilidad y adaptación necesarias para la CdS. Si bien ha habido iniciativas conjuntas en SOA y BPM o incluso a nivel inter organizacional, éstas se han enfocado en abordar los procesos de negocio como servicios interoperables con un enfoque top-down, por lo que elegido en lenguaje, la solución queda acoplada al mismo.

Concepto	Descripción
Prestación y entrega de servicios	Los servicios son proporcionados por unos actores a otros que los componen en procesos más complejos
Reutilización de servicios	Un determinado actor puede proporcionar el mismo servicio a muchos actores en la misma o en diferente red
Abstracción de un servicio	Los detalles de la implementación de los servicios que un actor ofrece están generalmente ocultos a otros actores, ya la implementación se asocia con el conocimiento que tiene un actor para agregar valor y es el elemento fundamental que da ventaja a un actor frente al resto en la red
Capacidad de descubrimiento de servicios	El conocimiento relativo a los servicios que pueden ser consumidos es público, por lo que los actores pueden ser identificados como potenciales miembros de la red
Contratación de servicios	Los servicios se adhieren a un acuerdo, que queda reflejado en uno o más contratos que regulan los términos bajo los que se realiza la prestación de los servicios
Autonomía y bajo acoplamiento	La lógica de negocio se encapsula en servicios con la intención de promover la reutilización. De esta forma, los autores tienen el control de la lógica encapsulada en los servicios. Los servicios débilmente acoplados dan soporte a la composición de los mismos para crear procesos más complejos con un conjunto de dependencias bien conocidas
Composición de servicios	Un proceso complejo proporcionado por un actor es el resultado de la composición de los servicios prestados por el actor en sí mismo u otros actores. Esta composición puede darse de dos formas: orquestación o coreografía. En la orquestación de servicios la colaboración entre los actores es controlada por uno de ellos que actúa como coordinador responsables de asignar actividades y supervisar su ejecución. En la coreografía los actores desarrollan procesos complejos de forma colaborativa, en el que la sincronización de sus esfuerzos se realiza en comunicaciones entre actores peer-to-peer
Monitorización de servicios	Los servicios proporcionados por los actores son constantemente monitorizados para analizar el cumplimiento de los términos contractuales, para lo cual se utilizan auditorías, protocolos de aceptación, indicadores clave, etc.

Tabla 1.2: Características de SOA con potencial aplicación a un entorno inter organizacional

1.3.5. Tendencias en la aplicación de MDE y SOC para la gestión de procesos de negocio

Anteriormente hemos citado MDE y SOC como dos tecnologías, provenientes de otros campos de la investigación, que pueden ser de aplicación al modelado de procesos de negocio con un enfoque inter organizacional como el que pretendemos. En esta sección analizamos las principales tendencias y líneas de trabajo propuestas en dichos ámbitos como refuerzo de la hipótesis y planteamiento del problema a resolver.

En 2007 Papazoglou y van den Heuvel [Papazoglou y Van Den Heuvel, 2007b] analizaron el estado de SOC y propusieron algunos temas abiertos para la futura investigación, en particular:

- Mejorar la flexibilidad a la hora de llevar a cabo la composición dinámica de los servicios, para lo que propusieron un enfoque basado en reglas de negocio.
- Mejorar en la transaccionabilidad a distintos niveles, desde los mensajes intercambiados hasta conversaciones entre sistemas, para lo que propusieron modelos basados en transacciones a nivel de procesos de negocio.
- Otros campos como el desarrollo de metodologías de soporte para el diseño de procesos de negocio con servicios y la refactorización de aplicaciones existentes para que pudieran ser integradas en el paradigma de procesos de negocio dirigidos por los servicios.

En 2008 Papazoglou et al. [Papazoglou *et al.*, 2008] establecieron la hoja de ruta de investigación en el dominio de SOC, organizados en cuatro niveles: fundamentos básicos sobre los servicios, composición, gestión e ingeniería de servicios. En cuanto a las relaciones entre SOC y BPM, apuntaron como elemento clave para su adopción en la industria el reto de ser capaces de dar soporte a la composición automática de procesos de negocio distribuidos, en los que la flexibilidad y la adaptación dinámica de los modelos de proceso dirija la composición de los servicios asociados. Para ello, los autores propusieron, partiendo de los existentes BPEL y WS-CDL, disponer de formas más avanzadas de coordinación, ser capaces de modificar instancias de modelos de procesos, dar soporte a modelos de procesos menos estructurados y establecer técnicas automáticas de planificación como parte de procesos integrados o compuestos. Apuntaron la falta de lenguajes que definan tanto las necesidades de las organizaciones a nivel de negocio, como sus requisitos de interacción con servicios proporcionados por otras organizaciones, disponiendo de un método sistemático que una los servicios con los procesos de negocio. De igual forma propusieron diferenciar entre procesos y servicios del negocio, apuntando a la necesidad de disponer de metodologías que integren el desarrollo de software tradicional con las técnicas de modelado de procesos de negocio, utilizando notaciones como BPMN o marcos como SCOR.

En 2007 Kamoun [Kamoun, 2007] apuntó un conjunto de factores clave para conseguir la adopción de BPM y SOA de forma integrada, entre los que destacamos: la necesidad de disponer de estándares reconocidos y adoptados por la industria, que combinaran bajo un mismo paraguas las disciplinas de BPM y SOA, así como comenzar por aquellos procesos de negocio pequeños, pero relevantes, que requieran gran flexibilidad y adaptación.

OASIS define el ecosistema SOA como *la red de procesos discretos y máquinas que, junto a una comunidad de personas, crean usan y gobiernan servicios específicos y proveedores externos de recursos que son necesarios por dichos servicios* [Estefan *et al.*, 2009]. SOA es uno de los paradigmas más utilizados en el ámbito de la SCM [Picard *et al.*, 2014], especialmente en las PYMEs, dando soporte a diversas estructuras organizativas como las comunidades de práctica [Lave y Wenger, 1991], los equipos virtuales [Lipnack, 1997], las empresas virtuales [Davulcu *et al.*, 1999] y las organizaciones virtuales [Ahuja y Carley, 1998].

En un entorno globalizado y competitivo, la sostenibilidad de las organizaciones es el mayor problema. Los entornos en los que la colaboración inter organizacional se desarrolla con el soporte de herramientas basadas en servicios, se consigue una mayor competitividad, reduciendo errores y ayudando por tanto a la sostenibilidad de las organizaciones [Picard *et al.*, 2014].

En 2010, Ziemann en su tesis doctoral [Ziemann, 2010] aborda la definición y ejecución de procesos colaborativos, mediante la propuesta de una arquitectura de interoperabilidad de sistemas de información. Para ello el autor propone una interfaz de interoperabilidad de procesos con un enfoque basado en modelos, siguiendo una aproximación de arriba a abajo, donde los posibles procesos de negocio existentes ya modelados no pueden ser reutilizados.

En 2012, Reichert y Webber [Reichert, 2012] analizan el campo de la flexibilidad de procesos de negocio. El soporte a la flexibilidad en los procesos entre organizaciones se presenta como uno de los retos a abordar de forma que puedan emerger cadenas de valor dinámicas, formadas a partir de diversas organizaciones, que tengan capacidad de implementar de forma eficiente nuevos procesos colaborativos que se adapten a los ya existentes para dar respuesta a las nuevas necesidades. Los autores identifican la necesidad de reutilizar los procesos existentes para evolucionarlos a cada caso de colaboración particular como un principal reto no resuelto.

MDE se ha convertido en la tecnología de base para establecer modelos de procesos [García-Borgoñón *et al.*, 2014] y como veremos en el capítulo 2, existen algunas propuestas basadas en modelos de aplicación al modelado de procesos de negocio de la CdS. A pesar de que en la literatura hay estudios que apuntan el interés futuro en la integración de las TIC o la adopción de un enfoque basado en modelos [Min y Zhou, 2002] [Perego *et al.*, 2011] [Pires y de Camargo Junior, 2010], en los trabajos existentes no han usado el potencial que el paradigma *MDE* tiene para describir procesos a través del metamodelado y de la transformación entre modelos.

Como principal conclusión podemos destacar que el dominio de la CdS, tiene las características anteriormente enunciadas para poder obtener los beneficios de un enfoque basado en servicios, como son procesos de negocio relevantes que requieran flexibilidad y adaptación. Por otro lado, su complejidad y la necesidad de interacción y gran diversidad de lenguajes de modelado, hacen de MDE un paradigma idóneo para, mediante modelos y transformaciones, disponer de modelos de procesos de negocio flexibles y adaptables.

Tras este contexto pasamos a exponer el problema a resolver en nuestra investigación.

1.4. Problema a resolver

Una vez hemos definido los principales términos del dominio de la CdS en el que vamos a aplicar nuestro trabajo, así como el contexto del modelado de los procesos de negocio junto con las tecnologías de soporte como SOA y MDE, en esta sección pasamos a enunciar las bases del planteamiento del problema a resolver. Partiremos de la descripción de la hipótesis de trabajo, continuaremos con el contexto de aplicación, que precede a la exposición de la necesidad detectada. Finalmente enunciaremos el esquema de la solución propuesto en este trabajo de tesis.

1.4.1. Hipótesis

Desde principios de los 80 se han desarrollado muchas iniciativas para resolver problemas específicos en la *SCM* mediante la aplicación de diversas técnicas para analizar, simular u optimizar operaciones, tácticas o estrategias. En todas ellas se parte de la creación de un modelo que simplifica la realidad y que permite analizar o construir la solución para el problema en particular.

El modelo siempre existe, lo que varía es su forma:

- Puede que sea un modelo mental, que únicamente exista en la cabeza de la persona que intenta solventar el problema.
- Puede que se haga explícito, para que potencialmente pueda ser compartido con otras personas.
- Puede que además de explícito, esté basado en algún formalismo, lo que permite ser interpretado de forma automática por sistemas o algoritmos. Ésta es la premisa sobre la que se basa *MDE*.

El enfoque basado en modelos ha podido ser aplicado de forma práctica en organizaciones individuales, pero cada vez más existe una fuerte dependencia entre las decisiones que una empresa toma y las que se determinan por otras organizaciones, clientes o proveedores, con los que forma una cadena de valor. Por ello es preciso disponer de modelos de un nivel jerárquico superior que nos permitan tener una visión holística de la CdS para crear soluciones que optimicen una actividad que es llevada a cabo por varios participantes.

La hipótesis de la que partimos en este proyecto de tesis es la siguiente: *MDE es un paradigma válido para modelar procesos con un enfoque inter organizacional y puede ser utilizado para convertir el modelado de procesos en una herramienta de apoyo a la toma de decisiones colaborativa en la CdS a nivel práctico.*

1.4.2. Problema a resolver

Hemos visto que las organizaciones definen sus procesos de negocio, de forma individual, pero luego en el mundo real interactúan con otras organizaciones, clientes y proveedores, formando cadenas de valor empresarial comúnmente conocidas como CdS. Para definir sus procesos de negocio, las organizaciones eligen de entre la gran cantidad de lenguajes de modelado existentes, apostando por un método para rediseñar, monitorizar o incluso ejecutar sus procesos de negocio individuales. Para poder tener una vista de los procesos de negocio a lo largo de toda la CdS, lo habitual es generar desde cero un nuevo modelo de proceso de negocio que incorpora las actuaciones llevadas a cabo por los diversos participantes. Este enfoque obliga a una compañía a duplicar el trabajo y a mantener la consistencia entre esa visión individual de su actividad y la integrada en cada CdS en la que participe.

Por otro lado, las CdS son muy dinámicas, entran y salen participantes, hay nuevos comportamientos y políticas, las decisiones se toman de forma distribuida, por lo que es preciso disponer de modelos de procesos flexibles y adaptables. Evolucionar los modelos de procesos de la CdS es un requisito imprescindible. Si lo hacemos sin considerar los cambios en los procesos individuales lo aleja de la realidad, ya que cada organización seguirá manteniendo sus decisiones e información de forma autónoma condicionando las operaciones en el resto de la cadena.

Para abordar el modelado de las CdS existen dos grandes enfoques, el top-down y el bottom-up [Shapiro, 1999], que han sido típicamente abordados al problema de la planificación y previsión de la demanda [Lapide, 2006].

- Un enfoque **top-down**: en el que se definen los procesos, sistemas y flujos de información a nivel global y a partir de los mismos cada organización que forma parte de la cadena adapta su funcionamiento interno. Este modelo es apropiado en CdS con poco dinamismo, en las que existe un gran actor que domina la CdS (una gran organización con toda su cadena de proveedores) y en la que los actores secundarios tienen pocas opciones de influir en la CdS o participar en otras CdS.
- Un enfoque **bottom-up**: en el que el proceso de la CdS emerge a partir de las vistas de cada organización en el proceso, que son adaptadas para favorecer la colaboración mediante el intercambio de información y la toma de decisiones de forma coordinada. Este modelo es apropiado para CdS formadas por pares iguales, que pueden formarse de forma temporal, en las que surgen nuevos actores y en las que la colaboración resulta esencial para aportar valor al cliente.

La figura 1.10 representa los dos posibles enfoques de modelado. En el contexto de este trabajo de tesis, el enfoque top-down implicará partir de la descripción del proceso colaborativo global y, a partir del mismo y de las necesidades de información y colaboración definidas, que cada organización ajuste sus procesos de trabajo internos al global. Apelando a la terminología SOC, el proceso global será una **orquestración** de los procesos individuales, que serán públicos a todas las organizaciones. En el caso de que una compañía participe en varias CdS, tendrá que adecuar sus procesos para mantener la coherencia con todas las CdS en las que participa,

siendo por tanto una posible barrera a dicha colaboración. De igual forma, este enfoque no aporta flexibilidad al modelo global, pudiendo suponer una barrera de entrada a nuevos actores. En definitiva, se trata de un enfoque válido cuando las relaciones entre pares no son iguales, habiendo un actor claramente protagonista, y cuando el dinamismo y la flexibilidad no son requeridas.

El enfoque bottom-up permite construir CdS temporales, muy dinámicas, mediante la integración de los procesos, sistemas y flujos de información de organizaciones a partir de los elementos que dichas compañías quieren compartir, que a su vez provienen de sus procesos de trabajo individuales. Esto permite que cada organización se centre en cómo aporta valor, sin modificar su proceso, sino creando interfaces que permitan su integración dinámica con otros actores, manteniendo la privacidad y autonomía en aquellas partes del proceso que considere. Atendiendo al lenguaje de SOC, en este enfoque cada organización ejecuta parte de un proceso como si de una **coreografía** se tratara.

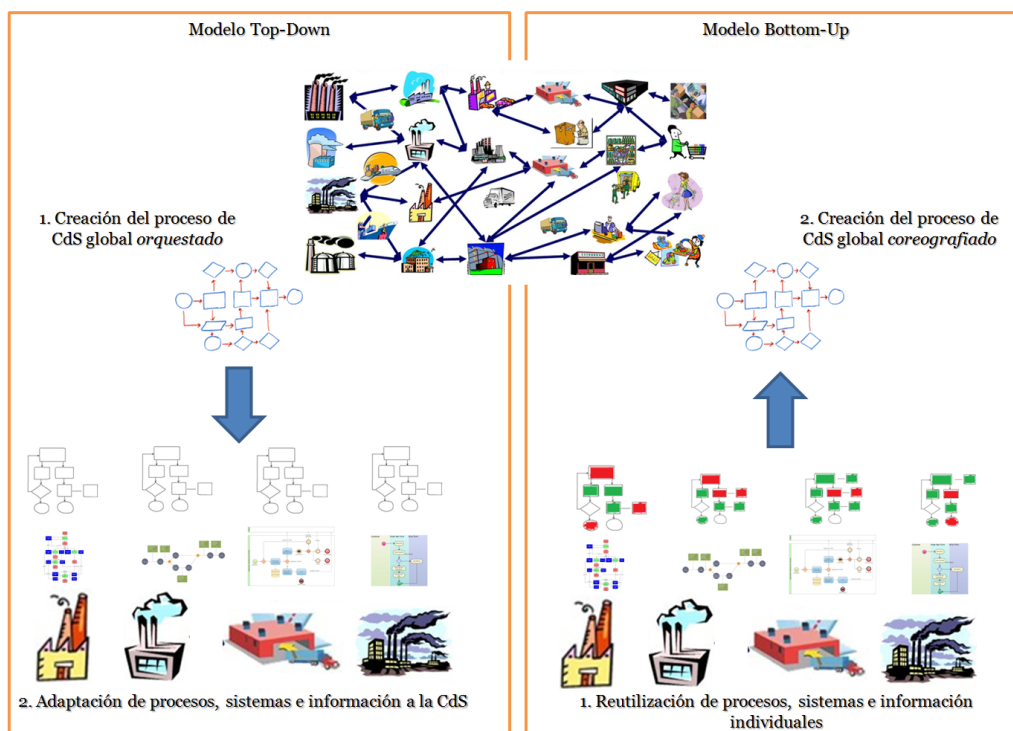


Figura 1.10: Enfoques top-down y bottom-up para el modelado de la CdS

Para crear estos modelos, existen multitud de lenguajes de modelado, con sus ventajas e inconvenientes y su diverso grado de adopción en la industria. Lo habitual es que en una CdS existan organizaciones que cooperan y que sus procesos de negocio están modelados con lenguajes diferentes, lo que dificulta su interacción y representación conjunta. Hasta ahora las diversas iniciativas han propuesto un enfoque top-down en el que se define el proceso general de la CdS y posteriormente se van enlazando los procesos individuales, para lo que se han propuesto diversos lenguajes de modelado, tal y como detallaremos en el capítulo 2, pero en este caso todas las organizaciones deben mantener un mismo lenguaje para modelar tanto su operativa interna

como la colaboración en la CdS.

Pero, ¿por qué obligar a todas las organizaciones a crear modelos de procesos de negocio con un mismo lenguaje?. Si además para poder llevar a la práctica de forma exitosa una gestión de los procesos de negocio entre las empresas, primero es preciso que cada organización gestione sus procesos, de alguna manera cada participante tendrá definidos sus procesos de la forma que mejor se adapte a sus necesidades para optimizar la operativa interna, por lo que nuestro planteamiento pasa por abordar el modelado de procesos de negocio de la CdS con un nuevo enfoque, bottom-up, en el que se dé soporte a una diversidad de lenguajes y herramientas y se reutilicen los modelos existentes.

En definitiva, nos proponemos abordar el problema de modelar los procesos de negocio de la CdS a partir de los modelos de procesos de negocio individuales que una organización pueda tener, con cualquier lenguaje de modelado de procesos, de forma que añadamos las capacidades de flexibilidad y adaptación de los modelos de procesos necesarias en el dominio de la CdS. Nuestra propuesta se basará en conceptos de SOA y en el paradigma MDE para, a partir de modelos existentes, generar transformaciones que definan una capa de interacción de procesos a nivel inter organizacional, para posteriormente diseñar el proceso general de la CdS como un orquestador de los anteriores.

1.4.3. Esquema de la solución

Para abordar este problema, nuestra propuesta de solución se basa en un marco de referencia denominado *Collaborative Business Generation (CBG)*, que pretende solventar el modelado de procesos y casos colaborativos con un enfoque bottom-up, y que está formado por:

- Un metamodelo, denominado *CBGProcess*, para dar soporte al modelado de procesos de negocio colaborativos.
- Una transformación M2M para generar la vista colaborativa desde un modelo de proceso individual.
- Un método para, a partir de la vista individual de un proceso conforme a *CBGProcess*, definir el proceso colaborativo. Dicho método permite separar la vista individual de un proceso de una organización de la vista colaborativa, manteniendo por tanto la privacidad y la autonomía en la toma de decisiones y compartiendo aquellos elementos que cada organización quiera para poder conformar el proceso en colaboración.
- Un metamodelo, denominado *CBGCase*, que siguiendo el paradigma de gestión de casos soporta el modelado de casos en entornos colaborativos.
- Una transformación M2M que permite, a partir de un proceso colaborativo conforme a *CBGProcess*, generar automáticamente una base de modelo de casos colaborativos conforme a *CBGCase*. De esta forma se consigue añadir la componente dinámica a los procesos, realizando una conversión del dominio de procesos al dominio de casos.

- Un método para, a partir del modelo de casos colaborativos generado, especificar qué elementos forman parte de la vista estática y cuáles en la vista dinámica del caso. De esta forma dispondremos del modelo de caso colaborativo generado a partir del proceso colaborativo.
- Una herramienta de soporte para facilitar la creación de modelos y la ejecución de las transformaciones desde un punto de vista práctico.

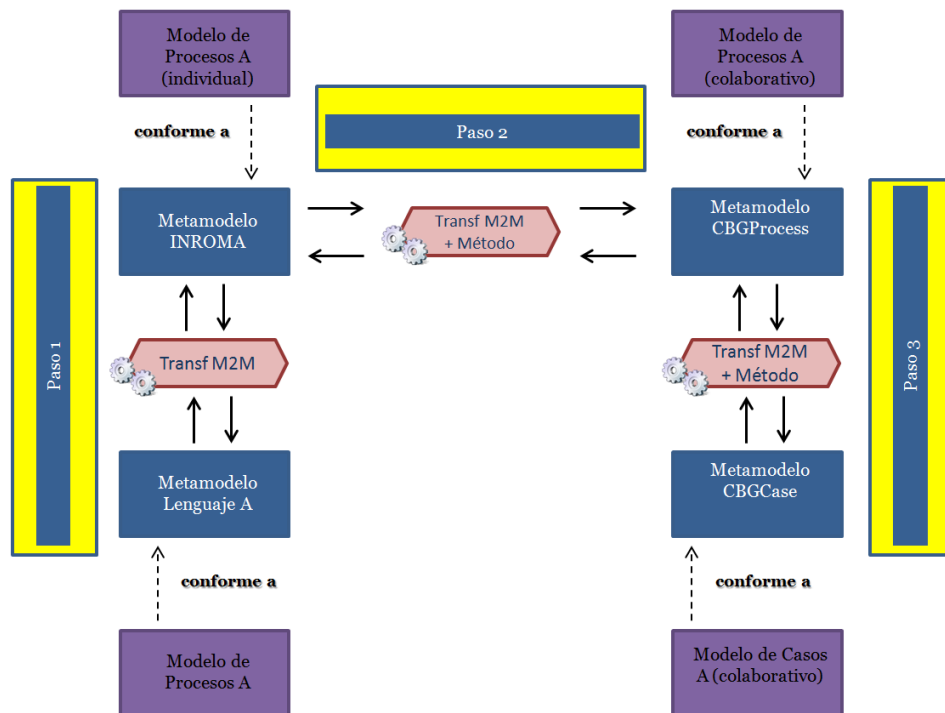


Figura 1.11: Resumen de la estructura de solución planteada en el marco CBG

Una vez hemos definido los elementos que van a formar parte de nuestra propuesta de solución, vamos a profundizar en cómo haremos uso de los mismos para abordar el problema planteado, tal y como se detalla gráficamente en la imagen 1.11. El detalle del planteamiento del problema y enfoque de la solución es presentado en el capítulo 3.

- Paso 1: Siguiendo un enfoque bottom-up, a partir del modelo creado en cada organización con cada lenguaje, estableceremos una descripción del mismo con el lenguaje de modelado de procesos *INROMA*.
- Paso 2: A partir del modelo individual, siguiendo el método para definir una colaboración entre diversos procesos y utilizando el metamodelo *CBGProcess*, se genera un modelo de colaboración de procesos de negocio. Como resultado hemos separado la vista individual de un proceso de una organización de la vista colaborativa, manteniendo por tanto la privacidad y la autonomía en la toma de decisiones y compartiendo aquellos elementos que cada organización quiera para poder conformar el proceso en colaboración.

- Paso 3: En el supuesto de que el escenario de la CdS requiera de añadir la componente dinámica a los procesos, a partir de una transformación M2M se genera automáticamente un modelo de casos colaborativos. Tomando como fuente este modelo de casos en colaboración, que ha sido generado a partir del modelo de procesos colaborativo, se debe utilizar el método planteado para especificar qué elementos forman parte de la vista estática y cuáles en la vista dinámica del caso. Como resultado dispondremos del modelo de caso colaborativo conforme a *CBGCase*.

1.5. Estructura de la tesis

A continuación se presenta la manera en la que se estructuran los siguientes capítulos en esta memoria de tesis siguiendo la siguiente pauta de exposición: en primer lugar estudiaremos en profundidad cuál es la situación actual en cuanto a las propuestas de lenguajes de modelado de procesos en la CdS, a partir de ahí, establecer los principales objetivos que se desean alcanzar en el desarrollo de este trabajo de tesis. Seguidamente, se expone y profundiza la propuesta de solución, abarcando tanto los aspectos más teóricos de la misma como las particularidades más prácticas. Así, el resto de esta memoria se constituye de la siguiente manera:

El capítulo 2 ofrece una visión general del estado del arte en relación a las diversas iniciativas existentes para modelar procesos en la CdS. Este análisis nos permitirá conocer con mayor grado de detalle las propuestas ya realizadas, los problemas que han intentado abordar así como las líneas de trabajo abiertas en la temática, de manera que tras este estudio podremos establecer de forma más concreta los objetivos específicos que podemos abordar.

Con la visión alcanzada después del estudio del estado del arte, en el capítulo 3 se establece el problema a resolver, planteando los aspectos que han determinado el talante con el que se afronta, el contexto en el que se ha desarrollado, así como las principales influencias que han determinado la solución propuesta, ofreciendo una primera vista de la misma.

A partir de aquí, una vez establecidos los objetivos derivados del problema a resolver, en los siguientes tres capítulos se va a profundizar en cada uno de los elementos en los que se sustenta la solución propuesta. Para ello, en primer lugar en el capítulo 4 se define una metamodelo *CBGProcess* para dar soporte al modelado de procesos de negocio colaborativos siguiendo un enfoque bottom-up reutilizando por tanto los modelos de procesos individuales. La propuesta se complementa con un método para, a partir de las vistas individuales de los procesos, definir el proceso colaborativo.

En el capítulo 5 se establece cómo el paradigma de gestión de casos puede ser utilizado para, de forma complementaria a la gestión de procesos, aportar las características de dinamismo, flexibilidad y adaptación presentes en los modelos de procesos en colaboración. En dicho capítulo exponemos la propuesta *CBGCase* para el modelado de casos colaborativos, junto al método para realizar el paso del dominio de procesos al de casos.

En el capítulo 6 se define cómo, aplicando las transformaciones en el ámbito de la ingeniería dirigida por modelos, se realiza la conversión tanto de las vistas individuales a las colaborativas como del dominio de procesos al de casos.

Todo este fundamento teórico se refrenda con la parte práctica de este trabajo de tesis, ya que todo el marco teórico no alcanzaría el impacto deseado en las organizaciones de software si para ello no contáramos con una herramienta que diera soporte a su automatización. Dicha herramienta, a la que hemos denominado *CBG-Tool*, se expone con detalle en el capítulo 7.

En el capítulo 8 se evalúa, mediante casos de estudios concretos extraídos de proyectos reales con la industria, que tanto la definición del problema planteado tiene lugar como que los resultados y las contribuciones resultado de este trabajo de tesis habrían facilitado el desarrollo de los mismos.

Además, en el capítulo 9 se reseñan, a modo de conclusiones, las aportaciones obtenidas con este trabajo de tesis, la forma como se han ido alcanzando los objetivos planteados, y las principales líneas de trabajo futuras derivadas de los resultados logrados.

Para finalizar, en los anexos se amplían algunos de los contenidos previamente presentados. En los anexos A y B se incluyen la totalidad de las transformaciones descritas mediante QVT Relations y enunciadas en el capítulo 6. En el anexo C se presenta un breve manual de la herramienta CBG-Tool desarrollada para dar soporte al marco teórico. Finalmente, en el anexo D se expone la trayectoria investigadora del autor de este trabajo de tesis.

1.6. Conclusiones

En este capítulo se ha introducido el contexto de la cadena de suministro y de su gestión, las diferencias con el dominio de la logística y las diversas interpretaciones de la colaboración o integración. Los principales marcos y modelos de referencia han sido enunciados con el objetivo de establecer los enfoques y alcances existentes para abordar el mismo problema. También se ha argumentado la necesidad de optimizar los procesos de negocio en la CdS, para lo cual se hace preciso disponer de un enfoque inter organizacional en el que un proceso es llevado a cabo por varias organizaciones que comparten un objetivo común. La dificultad de poner en práctica cambios en la CdS hace necesario tener entornos en los que los procesos puedan ser modelados y analizados, de forma que las mejoras puedan observarse de forma anticipada a su implementación en la realidad. Para poder generar modelos de procesos se utilizan lenguajes de modelado, existiendo una gran cantidad de alternativas en el campo de los procesos de negocio, ninguna aceptada como estándar de facto, así como algunas iniciativas en el dominio de los procesos de negocio de la CdS.

Establecido el contexto y marco del problema a resolver, hemos enunciado las principales teorías de la computación basada en servicios y la ingeniería del software dirigida por modelos y la forma en la que pueden ayudarnos a nuestro objetivo. La primera, ya que tiene un conjunto de características que hacen viable su adopción para el modelado de procesos de negocio inter organizacionales, en particular para llegar a cubrir las características de flexibilidad y adaptación necesarias para la CdS que han sido definidas. La segunda, por ser un paradigma que constituye la base de nuestra hipótesis de investigación.

Las diversas propuestas existentes, ya vengan del campo de SOA como de MDE, plantean un modelo top-down, partiendo del diseño del proceso global y posteriormente detallando el comportamiento individual, para lo cual todas las organizaciones deben acordar un único lenguaje tanto para modelar tanto su operativa interna como la colaboración en la CdS. Pero, ¿por qué obligar a todas las empresas a crear modelos con un mismo lenguaje?. Si además para poder llevar a la práctica de forma exitosa una gestión de los procesos de negocio entre las empresas, primero es preciso que cada organización gestione sus procesos, de alguna forma cada participante tendrá definidos sus procesos de la forma que mejor se adapte a sus necesidades para optimizar la operativa interna, por lo que nuestro planteamiento pasa por abordar el modelado de procesos de negocio de la CdS con un nuevo enfoque, bottom-up, en el que se dé soporte a una diversidad de lenguajes y herramientas y se reutilicen los modelos existentes. Por ello, hemos planteado el objetivo de este trabajo de tesis de cara a ofrecer una solución para mejorar la flexibilidad y adaptación de los modelos de procesos de negocio de la CdS. Por último se ha establecido una visión de cual va a ser la estructura de la memoria de esta tesis, en la que se expondrán los resultados obtenidos durante su realización.

Capítulo 2

Estado del arte

Como hemos expuesto en el capítulo anterior la presente tesis persigue proponer una solución para modelar los procesos de negocio de la CdS a partir de los modelos de procesos de negocio individuales que una organización pueda tener, con cualquier lenguaje de modelado de procesos, de forma que añadamos las capacidades de flexibilidad y adaptación de los modelos de procesos necesarias en el dominio de la CdS. El primer paso para alcanzar dicho objetivo es el conocer el estado de la situación actual en relación a las diversas iniciativas existentes para modelar procesos en la CdS. Este análisis nos permitirá conocer con mayor grado de detalle las propuestas ya realizadas, los problemas que han intentado abordar así como las líneas de trabajo abiertas en la temática, de manera que tras este estudio podremos establecer de forma más concreta los objetivos específicos que podemos abordar.

Por todo ello, este capítulo expone el estudio de la situación actual en los lenguajes de modelado de procesos de negocio de la CdS. En la primera sección, se presenta la metodología seguida para su elaboración, detallando sus resultados en la segunda sección. A continuación, en la tercera sección se realiza un análisis de cada una de las preguntas de investigación que nos hemos hecho. En la cuarta sección se apuntan, como principal contribución del estudio, las oportunidades detectadas como líneas de futura investigación abiertas. Finalmente, en la última sección de este capítulo se refieren las principales conclusiones del mismo como conclusión de las lecciones aprendidas de la revisión atendiendo al trabajo que tesis que nos ocupa.

2.1. Metodología para la realización del estado del arte

Para llevar a cabo este estudio apostamos por la guía propuesta por Kitchenham y conocida como revisión sistemática de la literatura (Systematic Literature Review, SLR), dado que es la más ampliamente aceptada en el campo de la ingeniería del software. Este método permite identificar, evaluar e interpretar todos los datos existentes para una pregunta de investigación (Research Question, RQ) en un área específica [Kitchenham y Charters, 2007], si bien recientemente han surgido discrepancias o propuestas de mejora sobre el método como veremos en la sección 2.1.2. Siguiendo las guías hemos desarrollado tres fases principales:

- Planificación, en la que hemos establecido las bases de lo que queremos obtener, cómo lo vamos a conseguir y la forma en la que lo documentaremos.
- Ejecución, en la que siguiendo una visión extendida de la metodología SLR hemos buscado, seleccionado, registrado y analizado los estudios.
- Reporting, en la que hemos documentado formalmente tanto el método como el estado actual y las líneas de trabajo abiertas.

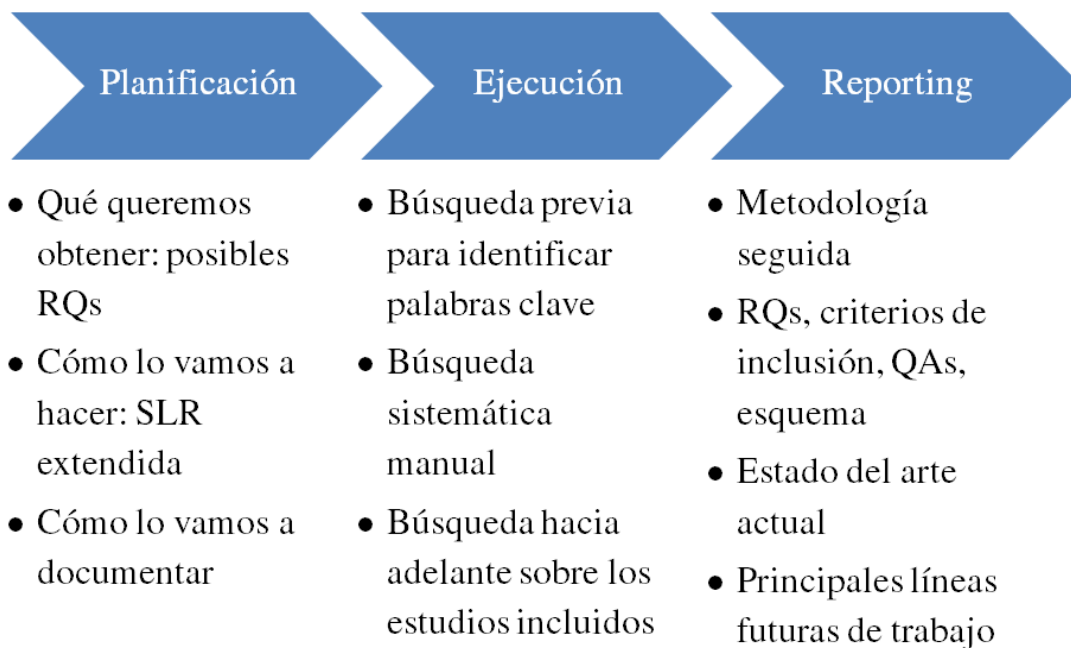


Figura 2.1: Fases principales de la metodología de la revisión sistemática de la literatura

Este esquema general se expone en la figura 2.1 si bien cada fase será detallada en las siguientes sub-secciones, en las que hablaremos de:

- Los antecedentes (sección 2.1.1) como el estudio previo realizado a la propia ejecución de la SLR.
- Las críticas existentes al método SLR que motivaron que siguiéramos en enfoque extendido, con una búsqueda previa y una posterior (sección 2.1.2).
- Los pasos definidos en el método SLR como son las preguntas (sección 2.1.3), la estrategia de búsqueda (sección 2.1.4), los criterios de inclusión y exclusión (sección 2.1.5), la forma en la que aseguramos la calidad de los estudios encontrados para las preguntas que nos hacemos (sección 2.1.6) y la manera en la que recogemos y analizamos la información de los mismos (sección 2.1.7).

2.1.1. Antecedentes y trabajos relacionados con el estudio de la situación actual

En esta sección vamos a presentar algunos estudios que han sido analizados para establecer el contexto de nuestra investigación y que constituyen los antecedentes del estudio del estado del arte que es expuesto en el presente capítulo. A la vista de los mismos, hemos decidido acometer una revisión sistemática de la literatura que nos permita actualizar las diversas revisiones y estudios publicados y que justifica la necesidad de realizar un nuevo estado del arte más orientado al contexto de los lenguajes de modelado de procesos que tengan aplicación en el ámbito de la CdS.

Trabajos en el dominio específico de SCM

Esta sección agrupa aquellos estudios que marcan el contexto dentro del dominio específico de nuestro trabajo, la CdS y su gestión entendida como SCM, con el objetivo de analizar los retos pendientes identificados por otros autores. Así en 2005 Stadler revisó los trabajos en el ámbito de la planificación avanzada y la SCM, para lo que apuntó la necesidad de colaborar para descentralizar la toma de decisiones en la CdS así como conseguir una integración de las diversas operaciones entre las organizaciones [Stadtler, 2005].

En 2002 Min y Zhou [Min y Zhou, 2002] analizaron el pasado, presente y futuro del modelado de la CdS, con la principal conclusión de la falta de herramientas analíticas que permitan analizar y explotar las ventajas derivadas de la integración de los conceptos de la CdS. Entre los retos pendientes en el modelado de la CdS destacaron principalmente el desarrollo de modelos basados en las tecnologías de la información, de forma que la validación de los mismos pueda llevarse a cabo con soporte tecnológico mejorando la calidad de los datos y facilitando por tanto la toma de decisiones.

En 2009 Stock [Stock, 2009] resumió los avances en tres bloques: 1) el desarrollo de métodos y técnicas para analizar los procesos de la CdS; 2) el desarrollo de soluciones para resolver problemas concretos en la CdS, y; 3) la capacidad de medir los resultados de diversas tácticas o estrategias en la CdS. A su vez, apuntó como principal tarea a desarrollar el abordar la resolución de problemas con un enfoque de CdS, en lugar de utilizar la perspectiva individual de una organización.

En 2010 Ivanov y Sokolov [Ivanov y Sokolov, 2010] apuntaron tres principales líneas de investigación futuras: los conceptos del negocio, el modelado, y la ingeniería del software con las herramientas asociadas. A su vez identificaron malos entendidos entre los que destaca la visión de que las CdS surgen de la nada, cuando en la realidad casi siempre existen estructuras, relaciones entre organizaciones y procesos definidos siendo el principal reto la identificación, análisis y mejora de los procesos de negocio clave.

En 2010 Grubic y Fan [Grubic y Fan, 2010] analizaron seis propuestas de modelos de CdS basados en ontologías e identificaron nueve áreas de mejoras, entre las que destacan la vista estática, parcial y sólo a nivel estratégico de la CdS lo que aleja estos modelos de la realidad.

En 2011 Perego et al. [Perego *et al.*, 2011] analizaron la agenda de investigación en el ámbito de las tecnologías de la información y la logística, en la que incluyeron la integración de aplicaciones, la creación de modelos para evaluar el impacto de la adopción de sistemas y la falta de estudios empíricos.

En 2011 Arshinder et al. [Arshinder *et al.*, 2011] extendieron una revisión de 2008 [Kanda *et al.*, 2008] y revisaron el estado actual y las direcciones futuras de investigación en la coordinación en la CdS. Los autores establecieron una taxonomía de diversos problemas que pueden ser resueltos en la CdS a través de la coordinación y resaltaron la importancia de que los participantes de una CdS aprecien la importancia de la colaboración. Como principal reto apuntaron la necesidad de tener un enfoque holístico en la CdS, así como la posibilidad de modelar diversos tipos de colaboración en la CdS y analizar las implicaciones para las partes y para toda la CdS con la ayuda de técnicas de simulación.

En 2012 Noordin et al. [Noordin *et al.*, 2012] analizaron la importancia de la confianza para poder colaborar en la toma de decisiones en la CdS, destacando la falta de estudios empíricos.

En 2012 Badole et al. [Badole *et al.*, 2012] realizaron una revisión de 282 estudios en la temática, presentando una taxonomía de problemas en la CdS, de técnicas de modelado y apuntaron cinco oportunidades de cara al futuro: 1) la necesidad de disponer de modelos y herramientas multi-objetivo para mejorar la integración y coordinación en la CdS; 2) la incorporación de métricas que evalúen de forma práctica el rendimiento de una CdS; 3) la aplicación de las nuevas tecnologías para dirigir la ejecución de las actividades y procesos de la CdS; 4) la mejora en la gestión de productos perecederos en la CdS, y; 5) la necesidad de diseñar y desarrollar CdS humanitarias en el caso de catástrofes naturales.

En 2013 Kim et al. [Kim *et al.*, 2013] analizaron el campo de los sistemas de información en SCM para identificar que los trabajos son muy diversos, faltando base teórica para tratar con estudios cualitativos.

En 2014 Espósito y Evangelista [Espósito y Evangelista, 2014] analizaron los modelos de empresas virtuales a partir de 357 trabajos, identificando seis conceptos comunes en los diversos trabajos y otros cinco en los que existen diversos enfoques o adopciones. Igualmente analizaron las perspectivas de orquestación y coreografía, cuando existe una integración vertical u horizontal en los procesos de la CdS.

Trabajos en el dominio del modelado de BP

El segundo bloque está formado por aquellos trabajos relacionados en el ámbito del modelado de procesos de negocio, así como los lenguajes existentes para llevarlo a cabo.

En 2004 Aguilar-Savén [Aguilar-Saven, 2004] analizó nueve técnicas y herramientas de modelado de procesos de negocio, sus fortalezas y debilidades de cara al usuario y al modelador, así como su propósito para describir, tomar decisiones de diseño de los procesos, ejecución del proceso y soporte en la promulgación con sistemas informáticos. Su conclusión fue que todavía falta un marco general que establezca lo que una técnica de modelado de procesos de negocio debe incluir para que sea exitosa.

En 2005 Lucas et al. [Lucas *et al.*, 2005] identificaron más de 70 lenguajes de modelado de procesos, concluyendo que hasta ese momento ninguno de ellos había sido aceptado por la industria como estándar.

En 2010 Mili et al. [Mili *et al.*, 2010] analizaron seis lenguajes de modelado de procesos tradicionales, tres orientados a objetivos, cuatro con capacidad de representar el dinamismo y tres orientados a la integración de los procesos. Como resumen analizaron en dos tablas el grado de cobertura de cada propuesta para representar las vistas organizativa, dinámica, funcional y de información, así como sus capacidades descriptivas, de análisis y de promulgación. Además de plasmar la gran cantidad de lenguajes existentes, los autores argumentaron que la mayoría de las propuestas nacieron con la intención de crear un estándar, algo que finalmente no se ha llegado a alcanzar.

En 2013 La Rosa et al. [La Rosa *et al.*, 2013] analizaron los diversos lenguajes que dan soporte al modelado de procesos de negocio personalizables, creando familias de modelos que varían entre sí mediante la adición o eliminación de fragmentos de proceso, atendiendo a lo que se conoce como la variabilidad de los procesos. Como principal conclusión obtuvieron que, a pesar de que existían múltiples lenguajes, faltaban herramientas de soporte así como estudios empíricos que los validaran.

En 2014 Malekan et al. analizaron la evolución de los lenguajes de modelado de procesos de negocio para dar soporte a la colaboración entre organizaciones. Tras identificar las necesidades de las redes colaborativas y los factores críticos de éxito, propusieron unos indicadores de requisitos clave que evaluaron entre las diversas propuestas para, concluir, con la afirmación de que BPMN parece que sea el mejor lenguaje con estos objetivos [Malekan y Afsarmanesh, 2014].

En 2016 Tarhan et al. [Tarhan *et al.*, 2016] revisaron los modelos de madurez de BPM, concluyendo que a pesar de que existen diversas propuestas, el nivel de evidencias empíricas que muestren la validez de estos modelos de madurez es todavía muy bajo.

En 2016 Cognini et al. [Cognini *et al.*, 2016] analizaron los trabajos relacionados en el área de lenguajes para el modelado de procesos de negocio flexibles, entendidos como la capacidad de adaptación en tiempo real de las instancias del proceso, así como las técnicas para facilitar la evolución de los mismos. Los autores concluyeron que la mayoría de los lenguajes dan soporte a la flexibilidad a través de la variabilidad, lo que obliga a conocer a priori todas las posibles estructuras de control del flujo de los elementos de los procesos. Como futuro identificaron la necesidad de un lenguaje de modelado que de forma nativa: 1) permita gestionar todas las posibles variantes en tiempo de diseño; 2) dé soporte a la desviación en tiempo real de las instancias de los procesos sobre el flujo definido, y; 3) facilite la evolución de los modelos de procesos, algo que es todavía un reto pendiente.

Trabajos en el dominio del modelado de la colaboración entre BP

Esta sección incluye aquellos estudios relativos con el modelado de coreografías de procesos como forma de representar la colaboración entre BP. En este punto debemos diferenciar tres posibles ámbitos de definición:

- Los procesos de negocio colaborativos, que describen las interacción entre dos entidades de negocio desde una perspectiva neutral, capturando todas las interacciones entre los partners [Ziemann *et al.*, 2007].
- Los procesos de negocio inter organizacionales, ejecutados por dos o más organizaciones autónomas, en las que al menos una de ellas expone su proceso interno como caja blanca para visualizar el flujo de control del proceso en colaboración [Grefen, 2009].
- Los procesos de negocio de la CdS, como una visión específica de colaboración inter organizacional, en el que se dan además de organizaciones distintas y autónomas, dependencias entre procesos internos y externos a varios niveles, ya que requiere de una coordinación vertical (entre cada empresa y la CdS) y horizontal (relacionando la producción, la logística, la gestión y la información) [Lambert *et al.*, 2005] [Xiaodong *et al.*, 2009]. Será esta tercera interpretación de colaboración de procesos de negocio la que dirija nuestro trabajo.

En 2001 Dayal et al. [Dayal *et al.*, 2001] analizaron el dominio del modelado de procesos colaborativos. En este estudio identificaron tres elementos clave para el uso efectivo de las definiciones comunes de procesos empresariales: 1) un metamodelo de proceso de negocio común y su lenguaje de esquema asociado, para que los procesos empresariales puedan codificarse de una manera estándar; 2) un mecanismo para la descripción común de procesos, incluyendo documentos comerciales que se puedan reutilizar fácilmente para procesos más complejos, y; 3) un mecanismo para que las empresas publiquen su disponibilidad para participar en procesos empresariales comunes, para que los socios potenciales puedan descubrirse automáticamente y participar en la ejecución del proceso.

En 2002 Wand et al. [Wand y Weber, 2002] propusieron una agenda de investigación en el campo de los sistemas de información y técnicas de creación de modelos conceptuales. Argumentaron que era preciso investigar el impacto de usar combinaciones de gramáticas para proporcionar representaciones de un dominio [Wand y Weber, 2002]. En este sentido, las coreografías son una técnica básica utilizada para lograr procesos inter organizacionales.

En 2007 Legner et al. [Legner y Wende, 2007] identificaron los principales retos pendientes en el diseño de BP inter organizacionales. Como necesidades de base identificaron la capacidad de exponer un proceso interno como una caja negra dentro de la colaboración, la autonomía y confidencialidad de los procesos individuales, así como la formalización de las interacciones entre los procesos. Apuntaron las capacidades de los enfoques basados en modelos para especificar interfaces, modelar inter dependencias y dar soporte a la ejecución de los procesos reduciendo la brecha entre el diseño y la implementación.

En 2011 Oliva et al. [Oliva *et al.*, 2011] realizaron una revisión sistemática de los estudios relativos a la adaptación de coreografías, con la conclusión principal de que la investigación en coreografía de servicios se encontraba todavía en sus etapas iniciales, donde la adaptación de las mismas no había sido explorada en profundidad.

En 2011 Riemer et al. [Riemer *et al.*, 2011] analizaron algunas herramientas de gestión de BP colaborativos con la conclusión de que si bien se había hecho un esfuerzo significativo en la investigación relacionada con la modelización de procesos, la principal preocupación se centró en la propia tarea de modelado y cómo dar apoyo a las personas con herramientas colaborativas en sus logros de modelado.

En 2014 Aleem et al. [Aleem *et al.*, 2012] revisaron diferentes enfoques de modelado de BP colaborativos. Argumentaron que para mantenerse competitivas en el mercado global, las empresas deben tener la capacidad de estandarizar, describir y adaptar la forma en que reaccionan a diferentes tipos de oportunidades de negocio. Identificaron cuatro enfoques de modelado diferentes: 1) los que derivan de extender UML; 2) los que provienen de extender redes de Petri; 3) las metodologías basadas en agentes y; 4) los métodos basados en la web semántica. Describieron algunos desafíos para el modelado colaborativo de procesos: 1) la representación de la concurrencia entre procesos y la sincronización de diferentes procesos en una organización; 2) la comprensibilidad del proceso de colaboración por personas no técnicas, los enfoques existentes carecen de simplicidad y estandarización, y; 3) la aplicación en la práctica porque la mayor parte de los estudios se basan en prototipos académicos.

Principales conclusiones identificadas en los trabajos previos

A la vista de los trabajos relacionados, resumimos a continuación las principales conclusiones que tomamos como punto de partida para esta revisión de la literatura:

- Sigue pendiente el reto de resolver los problemas con un enfoque de CdS, en lugar de utilizar la perspectiva individual de una organización [Stock, 2009].
- Las CdS no surgen de la nada, ya existen procesos, la clave está en identificar aquellos que son clave y mejorarlos [Ivanov y Sokolov, 2010].
- Existe una gran cantidad de lenguajes de modelado de procesos de negocio [Mili *et al.*, 2010] [Lucas *et al.*, 2005].
- No existen modelos y herramientas multi-objetivo para mejorar la integración y coordinación en la CdS [Badole *et al.*, 2012].
- No existen soluciones que permitan modelar diversos tipos de colaboración en la CdS [Arshinder *et al.*, 2011].
- BPMN parece que sea el mejor lenguaje para cubrir los objetivos del modelado de SCP [Malekan y Afsarmanesh, 2014].
- Se están desarrollando extensiones sobre los lenguajes de modelado para cubrir aspectos concretos del modelado de BP como la variabilidad [La Rosa *et al.*, 2013], la flexibilidad [Cognini *et al.*, 2016] o los modelos de madurez [Tarhan *et al.*, 2016], pero con poca evidencia empírica que permita validarlas.
- los avances han sido en los elementos a modelar, pero no en dar apoyo a cómo crear un modelo en colaboración a partir de las contribuciones de cada participante [Riemer *et al.*, 2011].
- La aplicación en la práctica porque la mayor parte de los estudios se basan en prototipos académicos [Aleem *et al.*, 2012].
- En la literatura se distingue entre procesos de negocio colaborativos [Ziemann, 2007], inter organizacionales [Grefen, 2009] y de la CdS [Lambert *et al.*, 2005] [Xiaodong *et al.*, 2009]. En nuestro trabajo tomaremos la tercera perspectiva en la que colaboran varias organizaciones distintas y autónomas, y además se requiere una coordinación vertical (entre cada empresa y la CdS) y horizontal (relacionando la producción, la logística, la gestión y la información).

Principales objetivos de la revisión de sistemática de la literatura

A la vista de las conclusiones anteriores, en los últimos años se han realizado numerosos estudios encaminados a mejorar ciertos aspectos en la SCM mediante la generación de modelos formales de ciertas actividades y su análisis, simulación y optimización, si bien estos trabajos se

han desarrollado desde la perspectiva de una organización individual y no han tenido un enfoque inter organizacional [Badole *et al.*, 2012].

Existen muchas propuestas de lenguajes para modelar BP [Mili *et al.*, 2010], si bien algunas de ellas adolecen de evidencias empíricas de su aplicación práctica [Aleem *et al.*, 2012]. Entre todas ellas, si bien en 2005 se hablaba de que ninguna estaba adoptada por la industria [Lucas *et al.*, 2005], quizás en la actualidad BPMN sea la más propicia para los objetivos que nos planteamos [Malekan y Afsarmanesh, 2014].

Pero el contexto de nuestro problema a resolver tiene una serie de características específicas que hacen que es posible que la solución pase por adaptar, extender o integrar alguna de las propuestas existentes, por lo que entendemos que se hace preciso profundizar en el estudio de la situación actual, motivo por el que hemos decidido abordar una revisión sistemática de la literatura.

En ella nos centraremos en los lenguajes para modelar procesos en el ámbito de la CdS, si bien vamos a particularizar el estudio para obtener las evidencias de si existen propuestas que cubran estas necesidades:

- Den soporte a la privacidad en los procesos individuales, estableciendo una parte pública y otra privada de los mismos.
- Permitan la autonomía en el diseño y ejecución de las tareas a nivel individual, desacoplando la vista de una organización de la global de la CdS.
- Ayuden a definir claramente los interfaces entre los participantes, sus procesos y el SCP.
- Permitan el modelado multi organizacional, estableciendo una clara responsabilidad sobre cada una de las tareas.

Adicionalmente, de las diversas propuestas, queremos analizar los siguientes elementos:

- Si dan cobertura a la consistencia entre los modelos intra e inter organizacional.
- Si soportan el modelado con múltiples perspectivas, esto es, que cada compañía pueda tener una vista diferente de la misma colaboración en la CdS.
- Si se pueden reutilizar de algún modo los modelos individuales de los BP, esto es, si el enfoque de modelado es *top-down* o *bottom-up*.
- Si en el escenario de modelado de la CdS se permite que cada organización pueda hacer uso del lenguaje de modelado de BP que quiera, o todos los participantes deben tener el mismo BPML.

Todos los estudios anteriormente citados, serán incluidos dentro de la fase de pre-búsqueda y formarán parte del grupo de estudios potenciales sobre los que aplicaremos los criterios de inclusión definidos.

2.1.2. Metodologías para desarrollar una revisión sistemática de la literatura

Como hemos comentado, la metodología SLR surge en el ámbito de la ingeniería del software en el año 2007 de la mano de Kitchenham [Kitchenham y Charters, 2007], siendo el enfoque más adoptado desde entonces para realizar estudios del estado del arte en esta disciplina. Desde entonces, y debido a su gran adopción por parte de la comunidad científica, han surgido propuestas de mejora sobre las guías iniciales que resumimos a continuación, centradas tanto en la evaluación de la calidad de los estudios incluidos como en los métodos de búsqueda.

Tres años después la propia Kitchenham [Kitchenham *et al.*, 2010] analizó el uso de las guías por parte de la comunidad concluyendo la falta de evaluación de la calidad de los estudios primarios incluidos, misma conclusión obtenida por Da Silva *et al.* [Da Silva *et al.*, 2011] al año siguiente.

Pero las principales debilidades del método han ido enfocadas a los procesos de búsqueda, cómo y dónde buscar los estudios primarios, ya que condiciona en gran medida los resultados obtenidos siendo repetitiva esa posible limitación en las diversas revisiones publicadas. En 2011 Zhang *et al.* [Zhang *et al.*, 2011] propusieron mejorar la búsqueda añadiendo aquellos estudios previamente conocidos o encontrados no directamente en la búsqueda manual (concepto que definen como “quasi-gold standard”, QGS) así como una mejora en la evaluación del rendimiento de la búsqueda (“quasi-sensitivity”). Dos años después estos mismos autores [Zhang y Ali Babar, 2013] defienden la importancia de las revisiones sistemáticas de la literatura de forma empírica, si bien advierten del necesario equilibrio entre seguir un método rigurosamente y el esfuerzo necesario para garantizar que todos los potenciales estudios han sido incluidos.

El 2013 encontramos otra propuesta de mejora sobre el método de búsqueda, Wohlin y Prikladniki [Wohlin y Prikladniki, 2013] proponen seguir un enfoque “bola de nieve”, de forma que una vez se ha seguido la propuesta de Kitchenham, aquellos estudios incluidos son revisados para analizar tanto los estudios que citan (ampliaría el conjunto con artículos anteriores) como aquellos por los que son citados (ampliación basada en artículos posteriores), pudiendo repetir este proceso recursivamente con el objetivo de encontrar estudios relacionados con la temática que con las palabras clave de búsqueda hubieran quedado fuera del listado inicial.

A la vista de todas las propuestas de mejora recibidas, Kitchenham *et al.* en 2013 [Kitchenham y Brereton, 2013] hicieron una revisión de las guías y sugirieron nuevas mejoras para las mismas, entre las que están no utilizar cadenas de búsquedas estructuradas y en su lugar añadir estudios primarios obtenidos de otras búsquedas (lo que Zhang *et al.* acuñaron como concepto “quasi-gold standard”).

Por todo ello, hemos partido de las guías propuestas en 2007 como base y hemos incorporado estas sugerencias de mejora para aumentar la cantidad de estudios obtenidos mediante iteraciones añadiendo, de los estudios incluidos, aquellos que son referenciados por o desde otros estudios, consiguiendo ese efecto de “bola de nieve” [Wohlin y Prikladniki, 2013]. La estrategia de búsqueda es detallada en la sección 2.1.4.

2.1.3. Preguntas del estudio

Modelar procesos de negocio no es un nuevo problema. A lo largo de los años ha habido diversas iniciativas que han propuesto lenguajes y soluciones para definir, representar, diseñar o simular procesos. Dado que nuestro trabajo se centra en el ámbito de los procesos de la CdS, el estado del arte persigue responder a una pregunta de investigación general: “¿Cual es el estado del arte en los lenguajes de modelado de procesos de negocio de la CdS?”. Dado que esta cuestión es muy abierta, para poder guiar el resto de la revisión sistemática de la literatura, podemos concretarla en las siguientes preguntas:

- RQ1. ¿Qué lenguajes de modelado de procesos de negocio de la CdS se han propuesto? Atendiendo a las necesidades anteriormente expuestas, de forma particular queremos analizar.
 - ¿Dan soporte a la privacidad de los procesos individuales?
 - ¿Permiten la autonomía de cada organización desacoplando su vista individual de la global?
 - ¿Definen los interfaces entre los participantes, sus procesos y el SCP?
 - ¿Soportan el modelado multi organizacional, incluyendo la responsabilidad sobre cada tarea?
 - ¿Dan cobertura a la consistencia entre los modelos intra e inter organizacional?
 - ¿Permiten el modelado con múltiples perspectivas?
 - ¿Se basan en un enfoque *top-down* o *bottom-up*?
 - ¿Permiten el uso de múltiples BPMLs?
- RQ2. ¿Qué problemas de la CdS se han intentado resolver?
- RQ3. ¿Cuáles son las limitaciones de la investigación actual?

2.1.4. Estrategia de Búsqueda

En esta sección vamos a detallar el método que nos ha permitido realizar una búsqueda exhaustiva en las principales librerías digitales para localizar aquellos artículos en revistas, congresos, conferencias y talleres que puedan ayudar a establecer el estado del arte en la temática que nos ocupa.

Atendiendo a la metodología para desarrollar una revisión sistemática de la literatura y a las mejoras propuestas que han sido expuestas en la sección 2.1.2, hemos seguido los conceptos de “quasi-gold standard” y del efecto “bola de nieve” para aumentar la calidad de los estudios seleccionados en nuestra revisión del estado del arte. Por ello la búsqueda ha sido realizada en tres fases como se muestra en la figura 2.2.

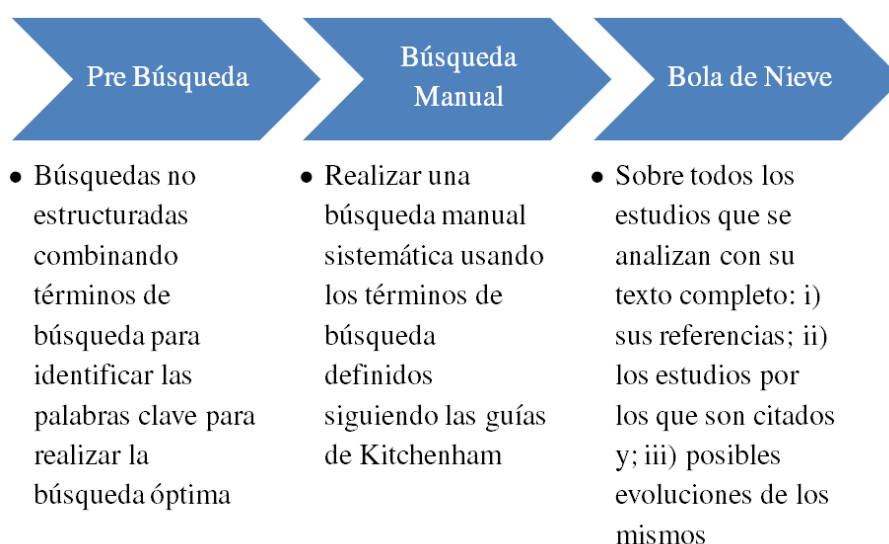


Figura 2.2: Fases principales del proceso de búsqueda de estudios primarios

Siendo conscientes de que las palabras de búsqueda condicionan en gran medida los resultados obtenidos, ha resultado vital la fase previa de análisis para identificar con claridad las principales palabras clave que debemos encontrar. En esta fase, hemos combinado un conjunto de términos y hemos realizado búsquedas para analizar a nivel empírico los resultados obtenidos y, a la vista de los mismos, ir seleccionando las mejores palabras clave para optimizar la búsqueda manual y equilibrar el método riguroso y el esfuerzo necesario. El listado inicial se muestra en la tabla 2.1, en la que cada concepto queda establecido por un conjunto de palabras y fuimos creando expresiones de búsqueda a partir de la combinación de los diversos conceptos representados por sus posibles términos.

Como resultado hemos obtenido los cuatro principales términos que definen y acotan el objetivo que perseguimos, analizar el estado actual en lo relativo a lenguajes para modelar procesos de la CdS. Por ello, los términos finales quedan enumerados por importancia de esta forma (se exponen en inglés dado que es el idioma en el que se ha realizado la búsqueda): 1) language; 2) supply chain; 3) process, y; 4) model.

Concepto	Palabras de Búsqueda
Lenguaje	language, DSL, domain specific, UML, unified modeling, BPMN, BPML, business process modeling language, notation
Cadena de Suministro	supply chain, logistics, integration, collaboration, negotiation, cooperation, inter organizational, decision making, VE, virtual enterprise, VO, virtual organization, CN, collaborative network, CNO, collaborative network organization
Proceso	process, business process, supply chain process
Modelo	model, meta-model, meta-meta-model, metamodel, MDE, model based, model-based, model driven, model-driven, MDA

Tabla 2.1: Diversas palabras de búsqueda que fueron utilizadas en la primera fase

Dado que los criterios para realizar la búsqueda, es decir, los campos sobre los que esos términos deben ser encontrados, también condicionan los resultados y, añadiendo el hecho de que los diversos buscadores integrados en las librerías digitales no ofrecen los mismos elementos sobre los que llevar a cabo el filtrado, hemos particularizado estos términos para cada buscador con el objetivo de minimizar el riesgo de excluir a priori estudios que pudieran ser de interés. Como norma general, hemos lanzado la consulta para encontrar todos los términos (language and “supply chain” and process and model) en todo el contenido del artículo, añadiendo la restricción del abstract o la unión de título, abstract y palabras clave cuando los resultados obtenidos eran miles. En el caso de que el número de estudios obtenidos fuera muy reducido, hemos intentado eliminar alguno de los términos por orden decreciente de importancia. La tabla 2.2 detalla los criterios de búsqueda establecidos para cada uno de las librerías seleccionadas. Debido al elevado número de estudios obtenidos hemos hecho uso de la herramienta Jabref [Alver, 2012] para registrar y manejar la información obtenida de una forma eficiente, seleccionada de entre otras alternativas por su capacidad de importar desde múltiples formatos [Basak, 2015].

Librería Digital	Criterios de Búsqueda
Google scholar	(language and “supply chain”) en el título
Science direct	(language and “supply chain” and process and model) en el título, abstract o palabras clave
Springerlink	(language) en el título y (“supply chain” and process and model) en todo el artículo
Scopus	(language and “supply chain” and process and model) en el título, abstract o palabras clave
IEEEExplore	(language and “supply chain” and process and model) en cualquier parte del artículo
ACM Digital Library	(language and “supply chain” and process and model) en el abstract

Tabla 2.2: Librerías digitales y criterios para realizar la búsqueda

Seguidamente, una vez finalizado el análisis de todos los estudios tras la búsqueda manual, para conseguir el efecto “bola de nieve” anteriormente comentado, para cada trabajo que se ha analizado en base a su texto completo, con independencia de si finalmente fue o no incluido, hemos realizado un análisis de las referencias de cada uno (para poder incluir aquellos anteriores en los que un estudio se basa pero que pudieron quedar fuera de los resultados de la búsqueda con las palabras clave), así como una búsqueda de aquellos estudios que, publicados con posterioridad, han citado al estudio incluido en nuestro estado del arte. Igualmente hemos extendido la relación con aquellos estudios que se basaban en lenguajes ya existentes, con el objetivo de acceder a los trabajos en los que ese lenguaje estuviera definido e integrarlos como un conjunto en nuestro estudio.

Por último, para aquellos estudios más relevantes que han definido mejor los siguientes pasos, bien por el nombre de la propuesta o por sus autores, hemos realizado consultas específicas de cara a identificar posibles trabajos nuevos y ampliar de esta forma el contexto de nuestro estudio. Para cada estudio recuperado hemos aplicado los filtros de inclusión y exclusión que se detallan en la siguiente sección.

2.1.5. Criterios de inclusión y exclusión

Uno de los pasos establecidos en las guías [Kitchenham y Charters, 2007] es el de establecer unos criterios objetivos para seleccionar, de los estudios primarios candidatos, aquellos que son incluidos para realizar el análisis del estado del arte. Dado que el método de búsqueda es empírico, se hace preciso trabajar de forma manual para profundizar y detectar si cada estudio puede contribuir en el trabajo de revisión sistemática en marcha.

Criterios de Inclusión	Criterios de Exclusión
Propone un lenguaje	Repetidos
Se dispone de su descripción formal (metamodelo)	Estudio no publicado
De aplicación al modelado de procesos	Idioma no inglés
En el ámbito de la CdS	Editorial, resumen de taller o panel
Aplica un BPML incluido en un SCP	No se consigue el texto completo
	Reglas sobre coreografías (e. GS1, HL7, etc)
	Modelos concretos de CdS orientados a su simulación

Tabla 2.3: Criterios de inclusión y exclusión en la fase de búsqueda

El proceso de selección parte de los estudios encontrados, tanto en la fase previa como en la búsqueda manual con las palabras clave anteriormente expuestas, para a la vista de los criterios establecidos en la tabla 2.3, puntuar: Sí, en el caso de que deba ser incluido; No, en el caso de que deba ser excluido; Parcial, cuando a la vista de la información existan dudas. Dado que en ocasiones la inclusión o exclusión de un estudio puede ser algo subjetivo, hemos elaborado esta fase en paralelo dos investigadores, sobre el título, resumen y palabras clave del artículo.

Posteriormente, en una reunión de consenso se analizan las diferencias y como resultado se obtiene un primer listado de estudios a incluir y a excluir. Aquellos estudios sobre los que sigue existiendo dudas, vuelven a ser analizados pero sobre el texto completo del artículo. La figura 2.3 expone esta primera fase y la figura 2.5 detalla el proceso para realizar el filtrado sobre el texto completo cuando en el paso anterior surgen dudas.

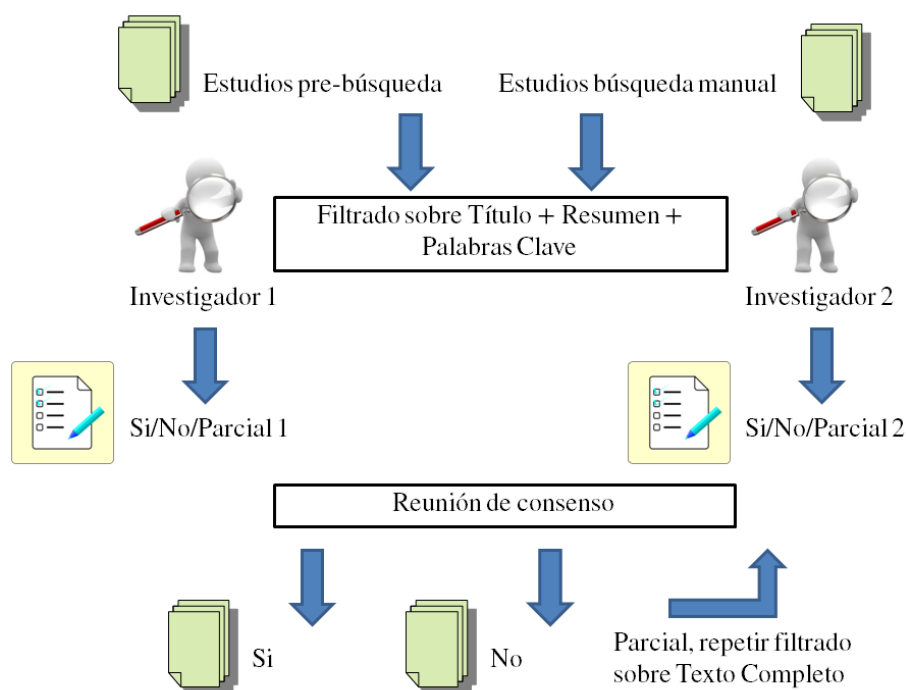


Figura 2.3: Proceso para seleccionar los estudios incluidos en la búsqueda inicial

Una vez que el análisis ha finalizado, sobre el conjunto de estudios que han sido analizados considerando su texto completo, se realiza la nueva búsqueda para encontrar las referencias de los mismos, aquellos estudios posteriores que citaron a los anteriores y posibles nuevos trabajos sobre las mismas propuestas o de los mismos autores. El proceso de selección de nuevo con este conjunto de estudios atendiendo a los mismos criterios de filtrado, como se muestra en la figura 2.4, volviendo a repetir el filtrado sobre el texto completo como se representa en la figura 2.5. Cabe destacar que el criterio de inclusión no se aplica individualmente a cada estudio o trabajo, sino que de forma recursiva se analiza en base a los diversos trabajos encontrados. De esta forma, aunque en un estudio se exponga una propuesta sin detallar el lenguaje, si a través de sus referencias somos capaces de encontrar otro trabajo en el que el lenguaje quede definido, ambos en su conjunto quedan incluidos como una única propuesta. Este método hemos debido adaptarlo dado que es habitual que un lenguaje de modelado aplicado a la CdS se encuentre reflejado en diversos trabajos, en unos el lenguaje, en otros las herramientas de soporte y en otros el caso de aplicación a la CdS.

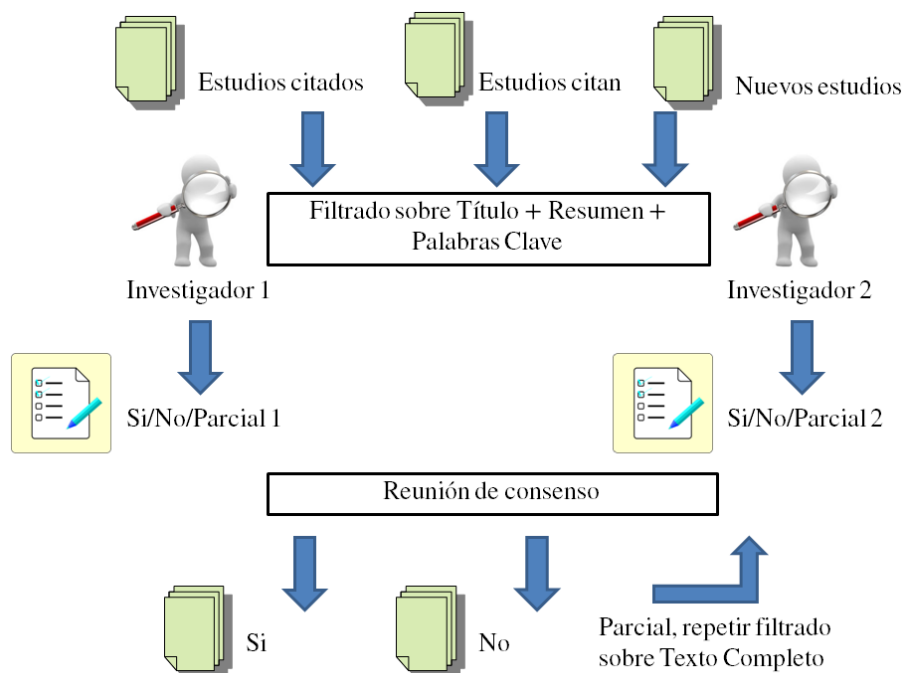


Figura 2.4: Proceso para seleccionar los estudios incluidos para crear la bola de nieve

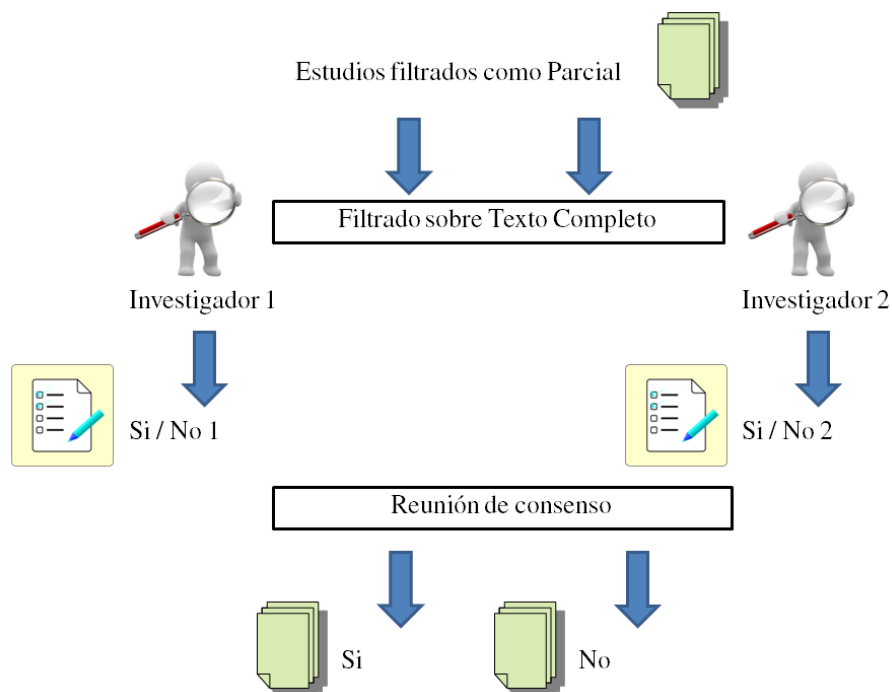


Figura 2.5: Proceso para discernir si un estudio es incluido a partir del texto completo

2.1.6. Aseguramiento de la calidad

Para evaluar la calidad de los estudios seleccionados para cumplir el objetivo que nos hemos planteado en este estado del arte hemos establecido un cuestionario que debe ser rellenado en cada uno. Para evaluar la RQ1, queremos valorar por un lado si cada propuesta analiza el trabajo realizado hasta la fecha, de forma que justifique hasta dónde había llegado la investigación actual y qué hueco pretendía cubrir, y por otro la propia descripción del lenguaje de modelado, algo que asociados a las preguntas QA1 y QA3 respectivamente. En relación a la RQ2 de nuevo valoraremos la puesta en contexto, en este caso en el ámbito de la CdS, de los problemas a resolver (QA2) si bien, dado que también queremos analizar su aplicabilidad en la práctica, vamos a analizar el método que en cada caso se ha llevado a cabo la validación de la propuesta (QA4). Por último, de cara a hacer una prospectiva del trabajo pendiente (RQ3), evaluaremos las líneas de investigación futuras planteadas en cada propuesta en la QA5. Para cada pregunta existen tres posibles respuestas, Sí, Parcialmente y No, tal y como quedan especificadas en la tabla 2.4.

Preguntas y criterios para analizar la calidad de los estudios incluidos

QA1: ¿El estudio realiza un estudio del estado del arte u otros trabajos relacionados?

- Sí: en el caso de que exponga los trabajos relacionados y enmarque su enfoque o propuesta en ese contexto
- Parcialmente: si sólo menciona algunos trabajos pero no establece el marco contextual en el que se desarrolla el estudio
- No: cuando no mencione trabajos relacionados

QA2: ¿Expone el estudio el problema que pretende resolver en el ámbito de la CdS?

- Sí: si explica la motivación y la necesidad de fondo en la CdS contra la que trabaja
- Parcialmente: cuando comente el problema pero no su motivación
- No: si no detalla el problema que pretende resolver

QA3: ¿El estudio, o sus referencias, detallan la propuesta de lenguaje para modelar procesos de la CdS?

- Sí: si se describe el lenguaje
- Parcialmente: cuando el lenguaje no queda completamente detallado
- No: si no describe el lenguaje

QA4: ¿Se presenta una validación de la propuesta?

- Sí: si se usa un caso real para la validación
- Parcialmente: cuando se utiliza un entorno simulado o de laboratorio para validar la propuesta
- No: si no se valida la iniciativa

QA5: ¿El estudio muestra los trabajos futuros de investigación?

- Sí: cuando se enumeran los siguientes pasos a dar
 - Parcialmente: si hay una descripción de posibles caminos futuros pero no concretos
 - No: si no se describen iniciativas futuras
-

Tabla 2.4: Preguntas para evaluar la calidad de los estudios incluidos

2.1.7. Recogida y análisis de información

Por último, para finalizar con los pasos previos definidos en las guías, hemos definido un esquema de información a modo de ficha en la que hemos recogido los datos de cada uno de los estudios incluidos en este estado del arte, teniendo como objetivo el facilitar el proceso de análisis de la información recopilada. La tabla 2.5 recoge la ficha para cada estudio.

Tipo de información	Datos almacenados
Información Básica	Título, autor y año de publicación
Datos de la Publicación	Libro, revista, conferencia en el que fue publicado, palabras clave, resumen y bibtex para su cita
Propuesta de Lenguaje	Nombre (si lo tiene), descripción, motivación, si se basa en otras propuestas o tecnologías de base, si está basado en modelos
Aplicación al dominio de la CdS	Problema que pretende resolver en el ámbito de la CdS), si tiene un enfoque inter organizacional, si soporta o no el modelado específico de los procesos con otros lenguajes, si el enfoque es top-down o bottom-up
Trabajo Relacionado	Cualquier otro estudio que pueda considerarse para su inclusión en este estudio del arte
Validación	Cómo los autores han validado la propuesta, en real, en laboratorio, sin validar
Referencias	Aquellas otras posibles citas del artículo que puedan ser incluidas en otra iteración, así como otros posibles estudios que lo citan o nuevas publicaciones de los autores en esta línea
Trabajo Futuro	Principales líneas de trabajo futuras tanto del artículo como en la temática, limitaciones actuales que ven los autores (si las hay)
Aseguramiento de calidad	Evaluación de las preguntas QA1, QA2, QA3, QA4 y QA5
Fase de inclusión	Para analizar la efectividad de la metodología, anotamos si el estudio ha sido incluido en la pre-búsqueda, en la búsqueda manual o a través de citas de terceros en la tercera fase que hemos denominado bola de nieve
Otros	Cualquier otra anotación que sea de utilidad para analizar las contribuciones del estudio en el estado del arte

Tabla 2.5: Esquema de información recogida para cada estudio incluido

2.2. Resultados del estudio del estado del arte

Tras presentar la metodología llevada a cabo para este estudio, en esta sección pasamos a mostrar los resultados obtenidos, mediante el detalle cuantitativo de los diversos trabajos encontrados en cada fase, como la evaluación de la calidad de los mismos.

2.2.1. Resultados de la búsqueda

En esta sección vamos a presentar a nivel cuantitativo los resultados obtenidos derivados de la ejecución de la metodología anteriormente expuesta. Tras la fase de pre-búsqueda, se fijaron las palabras clave y se recuperaron un conjunto de estudios que fueron directamente incorporados a su estudio. Seguidamente, en la fase de búsqueda se llevaron a cabo las búsquedas manuales y se fueron guardando todos los resultados. Para facilitar la gestión de las referencias, hemos utilizado la herramienta JabRef [Alver, 2012], que seleccionamos frente a otras alternativas por su capacidad de importar desde múltiples formatos [Basak, 2015]. Posteriormente, conforme se analizaban los estudios en base a su texto completo se identificaban nuevos trabajos candidatos para la fase iterativa que denominamos bola de nieve. La figura 2.6 muestra el conjunto de estudios que fueron analizados e incluidos en cada fase.

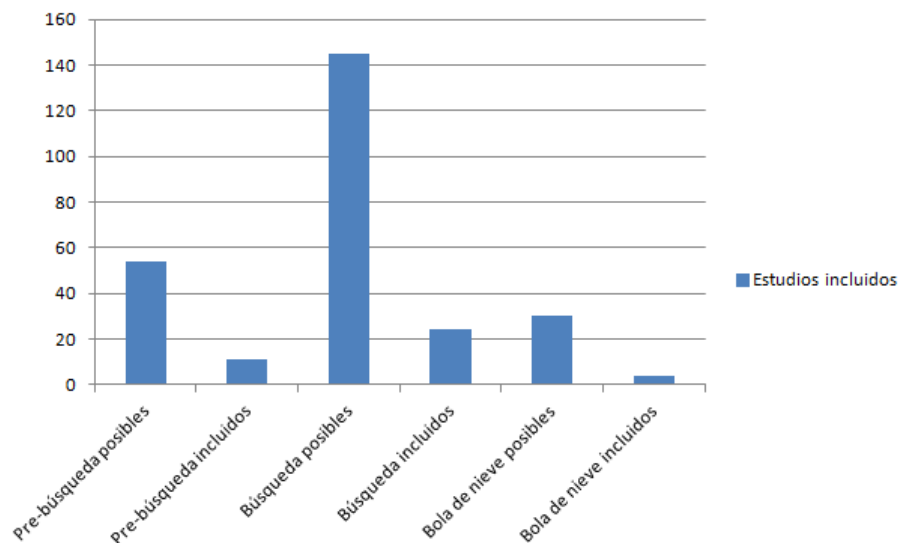


Figura 2.6: Número de estudios por fase

Si nos centramos en el resultado final de los estudios incluidos, la figura 2.7 nos muestra que han sido muy importantes tanto la fase previa, de la que hemos recuperado el 28 % de los trabajos, como la posterior con ese efecto de bola de nieve, que nos aporta un 10 % adicional, por lo que a nivel cuantitativo podemos concluir que las revisiones [Kitchenham y Brereton, 2013] realizadas sobre las guías iniciales [Kitchenham y Charters, 2007] mejoran los trabajos incluidos en el estudio.

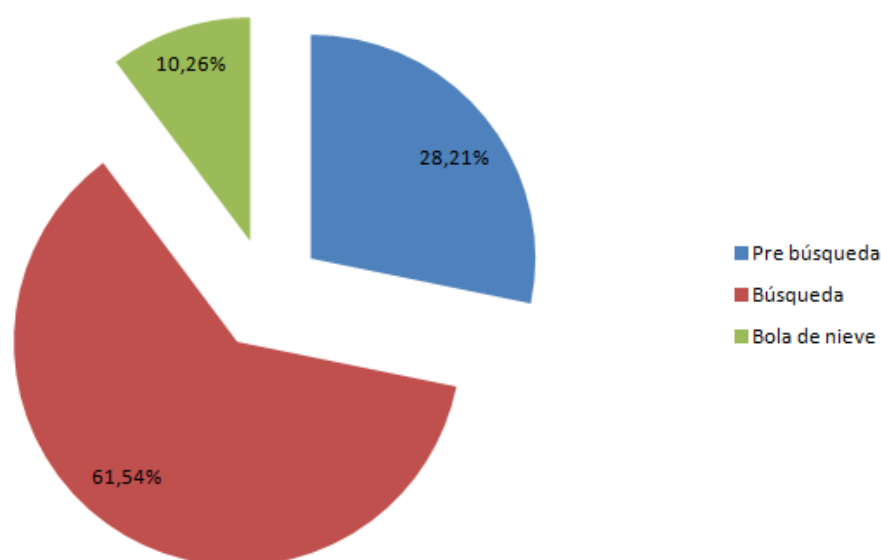


Figura 2.7: Porcentaje de estudios incluidos por fase

2.2.2. Evaluación de la calidad

En esta sección vamos a presentar los resultados obtenidos en relación a la evaluación de la calidad de los estudios incluidos, atendiendo a la tabla 2.4 anteriormente presentada. Dado que el método iterativo nos ha llevado a incluir diversos trabajos hasta encontrar la propuesta de lenguaje, el método de evaluación de la calidad no se ha realizado sobre cada trabajo individual sino sobre el grupo, es decir, realizamos una única valoración de todas las QAs en base al conjunto de estudios incluidos que hacen referencia a una propuesta de lenguaje de modelado de procesos de negocio en el ámbito de la CdS.

Para analizar cada una de las QAs de forma individual, hemos confeccionado la gráfica 2.8 en la que tenemos representado el porcentaje de Sí, Parcial o No, en base a todas las propuestas incluidas en el estudio. A la vista de esta imagen, podemos resaltar que:

- En relación a la RQ1, tanto el trabajo relacionado (QA1) como el detalle de la propuesta (QA3) están ampliamente cubiertos.
- En cuanto a la RQ2, en todos los casos el problema a resolver en el ámbito de la CdS está motivado (QA2), si bien la validación de las propuestas es algo que no queda cubierto en la mitad de las propuestas (QA4).
- En cuanto a la RQ3, de nuevo casi la mitad de las iniciativas quedan sin cubrir ese criterio de evaluación (QA5).

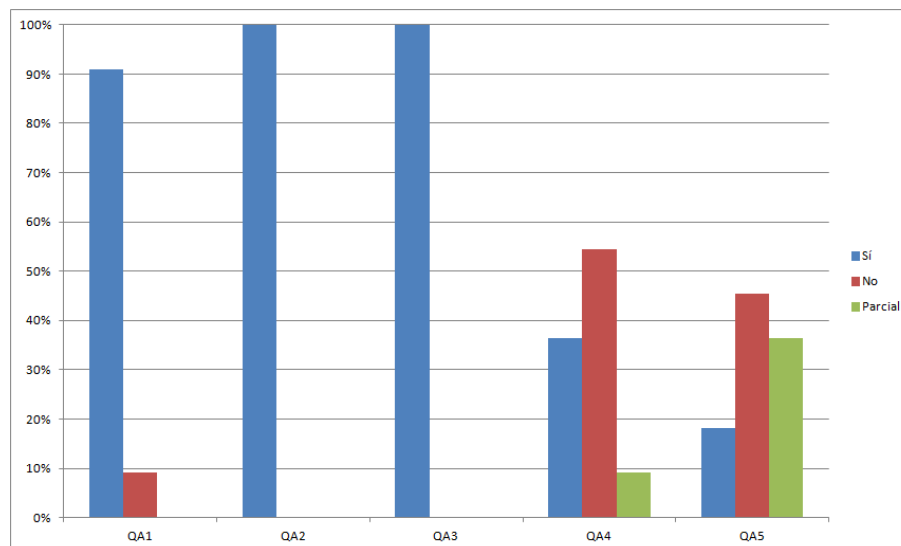


Figura 2.8: Evaluación de cada pregunta de aseguramiento de calidad en base a los estudios incluidos

Como principal conclusión de esta evaluación de calidad de los estudios incluidos, podemos destacar que casi la totalidad de los trabajos motivan el problema a resolver, justifican la necesidad de una nueva propuesta y detallan la misma, si bien podremos obtener menos evidencias en cuanto a su grado de validación y el trabajo futuro en la materia.

2.3. Análisis del estado del arte

En esta sección vamos a analizar el resultado de nuestra investigación sobre el estado del arte a la vista de los diversos trabajos incluidos. Para cada una de las tres preguntas de investigación (RQs) presentaremos y analizaremos los datos, finalizando con las principales limitaciones que ha tenido nuestro estudio.

2.3.1. RQ1: ¿Qué lenguajes de modelado de procesos de negocio de la CdS se han definido?

En primer lugar, queríamos saber los diversos lenguajes que han sido propuestos para poder modelar procesos de negocio en el ámbito de la CdS. Las tablas 2.6, 2.7 y 2.8 resumen las 21 iniciativas planteadas. Para cada una se presenta el año en el que se propuso, su nombre o acrónimo, en qué otros lenguajes o tecnologías de modelado se basa, la descripción de la misma y las referencias de los diversos estudios incluidos de las que se ha extraído la información. Destacamos que, debido a que hemos iterado en el proceso de búsqueda, en ocasiones una propuesta no se podía incluir dado que no se detallaba el lenguaje, pero al citar otro estudio en el que estaba, por ese motivo hemos ido iterativamente añadiendo más trabajos que quedan agrupados en propuestas. De igual forma, en el caso de que una misma propuesta haya evolucionado, se ha incluido la versión más reciente de la misma.

A la vista de los resultados, podemos observar que existen propuestas del año 2015, por lo que parece que es un tema no resuelto en el que la comunidad sigue trabajando. En cuanto a las tecnologías de base, más de la mitad hacen uso de UML, y el resto hacen extensiones o adaptaciones sobre lenguajes existentes como XML, SPEM, SCOR o SysML entre otros. En comparación con la revisión realizada en 2012 por Badole et al., en la que clasificaban los tipos de modelos en 10 tipos (programación dinámica, programación lineal, algoritmos genéticos, teoría de juegos, simulación, redes neuronales, redes de Petri, colonias de hormigas y modelos matemáticos) [Badole *et al.*, 2012], deberíamos añadir una nueva categoría de lenguajes basados en modelos, en la que entrarían la inmensa mayoría de estas iniciativas. Parece por tanto que estamos en una evolución del nivel de abstracción de los modelos de procesos de la CdS, donde lo relevante no es detallar la operativa sino entender y analizar las relaciones entre los participantes, para lo cual el paradigma basado en modelos está siendo el más utilizado.

Por otro lado podemos destacar que algunas de las propuestas incorporan resultados de investigación de otros ámbitos o dominios, no necesariamente relacionados directamente ni con la CdS ni con el modelado de procesos, como por ejemplo las arquitecturas de software, las ontologías o el campo de la ingeniería de sistemas. Esto pone en evidencia la fuerte relación de los sistemas de información en el campo de la CdS como un facilitador necesario a la hora de poder recoger datos que faciliten el análisis y la posterior toma de decisiones en la reingeniería de los procesos.

Año	Nombre	Basado en	Descripción	Referencias
1962	Redes de Petri (Petri Nets)	-	Creadas por Carl Adam Petri en su tesis doctoral, suponen un método formal para el modelado gráfico y matemático de sistemas de eventos discretos y concurrentes.	[Petri, 1962]
1992	EPCs (Event-Process Chains)	Redes de Petri	EPC es un lenguaje basado en Redes de Petri que permite incorporar roles y modelos de datos a través de la representación de los flujos de función, salidas e información asociados a un proceso. En 2006 está disponible como un perfil UML [Korherr y List, 2006]	[Scheer et al., 2005]
1995	IDEF3 (Integrated DEFinition Methods) process description language	-	IDEF3 ofrece un lenguaje y un método para documentar procesos, permitiendo que un mismo proceso sea modelado atendiendo a diversos puntos de vista.	[Mayer et al., 1995]
2002	UEML (Unified Enterprise Modelling Language)	EML	Propuesta que surge, no como un nuevo lenguaje en sí mismo aunque también lo sea, sino para proporcionar una interfaz uniforme a las diferentes herramientas de modelado así como un formato neutro para intercambiar modelos	[Vernadat, 2002]
2004	ECA (Enterprise Collaboration Architecture)	UML	EDOC (Enterprise Distributed Object Computing) es el perfil UML del OMG para dar soporte a la computación distribuida basándose en MDA y SOA. ECA es el metamodelo que define los roles interactúan dentro de las comunidades en el desempeño de los procesos de negocios colaborativos.	[OMG, 2004]
2006	UMM (UN/CEFACT'S modeling methodology)	UML	Se trata de una metodología de modelado desarrollada por UN/CEFACT (United Nations Center for Trade Facilitation and Electronic Business) con el objetivo de capturar los requisitos de los procesos de negocio inter organizacionales, para describirlos en un modelo independiente de su plataforma de ejecución. [Hofreiter et al., 2006]	
2006	ebXML BPSS (ebXML Business Process Specification Schema Technical Specification)	XML	ebBP es una especificación XML para describir procesos de negocio ejecutables e interoperables que surge con el objetivo de hacer de puente entre el modelado de BP y su ejecución mediante herramientas software.	[Dubray et al., 2006]

Tabla 2.6: Propuestas de lenguajes de modelado de procesos de la CdS existentes

Año	Nombre	Basado en	Descripción	Referencias
2006	PIM4SOA (Platform-Independent Model for Service-Oriented Architecture)	UML	Proponen desarrollar modelos de colaboraciones entre empresas que sean promulgables y ejecutables usando tecnologías de desarrollo basadas en modelos y en arquitecturas software	[Roser <i>et al.</i> , 2006]
2007	SCOL (Supply Chain Ontology Language) y SCML (Supply Chain Markup Language)	SCOR, IDEF	Proponen dos lenguajes para facilitar la integración de procesos de la Cds. SCOL, un lenguaje basado en ontologías, creado a partir de SCOR [SCC, 2014], para definir procesos en alto nivel. SCML como notación XML para su análisis cuantitativo. Para la descripción de los procesos a alto nivel hacen uso de IDEF y lo transforman a UML como paso previo a su análisis con sistemas TI	[Chandra, 2008], [Chandra, 2007]
2007	BPM for multi-enterprise cooperation	UML	Proponen un metamodelo basado en UML para representar los procesos de negocio, tanto el flujo interno de los mismos como la colaboración entre empresas de la Cds	[Hongxiu <i>et al.</i> , 2007]
2007	SUDDEN DSVL (Domain-Specific Visual Modelling Language)	EML	Hace uso de EML (Enterprise Modelling Language) para facilitar a las PYMES el modelado (y simulación) visual de sus Cds y compartir el conocimiento a nivel semántico	[Zhang <i>et al.</i> , 2010], [Zhang, 2010], [Li <i>et al.</i> , 2008], [Li <i>et al.</i> , 2007]
2008	VECCF (Virtual Enterprise Collaboration Framework)	UML	Mediante UML especifica un lenguaje para facilitar la colaboración entre las empresas mediante la mejora de la integración de sus diversos elementos y procesos a través del modelado de: los procesos de negocio, el software y las tecnologías	[Choi <i>et al.</i> , 2008]
2009	SCML (Supply Chain Modeling Language)	XML	Propone una metodología estándar genérica con información específica para el dominio de la Cds para dar soporte al modelado cuantitativo de la Cds	[Chatfield <i>et al.</i> , 2009]
2009	UMM for local choreographies	UMM	Extiende UMM (UN/CEFACT Modelling Methodology [Dietrich J, 2006]) para dar soporte al modelado de las relaciones de un participante de la Cds con tres o más miembros, lo que denomina las coreografías locales (local choreographies)	[Hofreiter, 2009]

Tabla 2.7: Propuestas de lenguajes de modelado de procesos de la Cds existentes (cont)

Año	Nombre	Basado en	Descripción	Referencias
2011	BPMN 2.0 (Business Process Model and Notation)	UML	Especificación estándar del OMG [?] para el modelado de procesos (privados y públicos) y sus coreografías	[OMG, 2010]
2011	IOBPM (inter Organizational Business Process Metamodel)	UML	Mediante UML proponen un lenguaje para modelar procesos de negocio entre organizaciones, siguiendo un enfoque de arriba a abajo (top-down) en el que se define el negocio (nivel organizacional), los conceptos (nivel lógico) y la parte técnica (nivel de ejecución)	[Bouchbout y Alimazighi, 2011]
2012	ArchMDE (Architecture-centric Model Driven Engineering)	UML	Hacen uso de ArchMDE [Azaiez, 2007] para modelar y analizar procesos de una CdS, especialmente en el ámbito de las PYMEs, creando un metamodelo para el dominio específico de la CdS que generaliza los conceptos, estructura y relaciones	[Tounsi <i>et al.</i> , 2012]
2014	SysPEM (System Process Engineering Metamodel)	SPEM y SysML	Combinan los conceptos claves de SysML (System Modeling Language [Friedenthal <i>et al.</i> , 2014]) con el modelado conceptual de SPEM (Software Process Engineering Metamodel [2007]) con el objetivo de definir, modelar y orquestar procesos	[Jakjoud <i>et al.</i> , 2014]
2014	SimulValor Extension	SimulValor	Realizan una propuesta de extensión sobre el lenguaje SimulValor [Daaboul <i>et al.</i> , 2012] para crear un marco y una herramienta que permita modelar, simular y analizar el valor de una red (en nuestro ámbito una CdS) para poder crear un sistema de apoyo a la toma de decisiones	[Daaboul <i>et al.</i> , 2014]
2015	AD UML 2.5 (Activity Diagrams, Unified Modeling Language)	UML	Los diagramas de actividad pueden ser utilizados para llevar a cabo el modelado organizacional de los procesos de negocio	[OMG, 2015b]
2015	CSSPEM (Collaborative Software Systems Process Engineering Metamodel)	SPEM 2.0	Propuesta que extiende SPEM para dar soporte al modelado y ejecución de procesos software colaborativos, mediante un nuevo elemento <i>CollaborationStructure</i> en la vista estructural, así como una nueva vista asociada al comportamiento para dar soporte a la ejecución	[Vo <i>et al.</i> , 2015]

Tabla 2.8: Propuestas de lenguajes de modelado de procesos de la CdS existentes (cont)

Seguidamente pasamos a analizar con mayor detalle cada una de las propuestas, atendiendo a las preguntas que nos hicimos en la RQ1 y que resumimos a continuación, cuyo resultado se muestra en la tabla 2.9, en el que resumimos con tres valores el análisis: 1) Sí, cuando tengamos evidencias de su cumplimiento; 2) No, cuando no las tengamos, y; 3) Parcial, cuando se pudiera llegar a cumplir sin incluirlo de forma explícita en el modelo. En cuanto al enfoque de modelado los valores serán: 1) Top-down, cuando sólo se pueda llevar a cabo ese enfoque; 2) Bottom-up, en caso contrario, y; 3) Ambos, si pudieran ser aplicados ambos enfoques.

- Nombre: nombre de la propuesta de lenguaje.
- Privacidad: ¿Dan soporte a la privacidad de los procesos individuales?. En este punto valoraremos la capacidad del lenguaje de incluir de forma explícita que los elementos sean públicos o privados.
- Autonomía: ¿Permiten la autonomía de cada organización desacoplando su vista individual de la global?. Aquí valoraremos la capacidad del lenguaje de soportar que cada organización diseñe y ejecute sus procesos de manera aislada al resto.
- Participantes: ¿Definen los interfaces entre los participantes, sus procesos y el SCP?
- Multi Organizacional: ¿Soportan el modelado multi organizacional, incluyendo la responsabilidad sobre cada tarea?
- Consistencia: ¿Dan cobertura a la consistencia entre los modelos intra e inter organizacional?. Aquí valoraremos la capacidad del lenguaje de establecer algún tipo de trazabilidad entre los elementos del proceso individual y del colaborativo, de forma que luego la organización pueda ser autónoma.
- Multi Perspectiva: ¿Permiten el modelado con múltiples perspectivas?. En este punto valoraremos la capacidad de que las organizaciones vean la colaboración atendiendo a diversos tipos de BP (procesos estructurados, dirigidos por los datos, casos, etc)
- Enfoque de modelado: ¿Se basan en un enfoque *top-down* o *bottom-up*?
- Multi BPML: ¿Permiten el uso de múltiples BPMLs?. Finalmente analizaremos si es factible lograr la interoperabilidad de modelos conformes a diversos BPMLs.

Nombre	Privacidad	Autonomía	Participantes	Multi Organizacional	Consistencia	Multi Perspectiva	Enfoque de modelado	Multi BPM
Redes de Petri	No	Parcial	Parcial	No	Parcial	No	Top-down	No
EPCs	No	No	Parcial	No	Parcial	No	Top-down	No
IDEF3	No	No	No	Parcial	No	No	Top-down	No
UEML	Parcial	Parcial	Sí	Sí	Parcial	No	Ambos	Parcial
ECA	No	Parcial	Parcial	Sí	No	No	Top-down	No
UMM	No	Parcial	Sí	Parcial	No	No	Top-down	No
ebXML BPSS	No	Parcial	Sí	Sí	No	No	Top-down	No
PIM4SOA	No	Parcial	Sí	Sí	No	No	Top-down	No
SCOL	No	Parcial	Sí	Sí	No	No	Top-down	No
BPM multi enterprise	No	Parcial	Sí	Sí	No	No	Top-down	No
SUDDEN DSVL	No	Parcial	Sí	Sí	No	No	Top-down	No
VECCF	No	Parcial	Sí	Parcial	No	No	Top-down	No
SCML	No	No	Sí	Sí	No	No	Top-down	No
UMM local choreographies	No	No	Parcial	Parcial	No	No	Top-down	No
BPMN 2.0	Sí	Parcial	Sí	Sí	No	No	Top-down	No
IOBPM	Sí	Sí	Sí	Sí	Parcial	No	Top-down	No
ArchMDE	No	No	Sí	Sí	No	No	Top-down	No
SysPEM	No	No	Sí	Sí	No	No	Top-down	No
SimulValor Extension	No	No	Sí	Sí	No	No	Top-down	No
AD UML 2.5	No	Sí	Sí	Sí	Parcial	No	Top-down	No
CSSPEM	No	Parcial	Sí	Sí	No	No	Top-down	No

Tabla 2.9: Análisis de las propuestas de lenguajes de modelado de procesos de la CdS existentes

A la vista de la información de la tabla 2.9, la figura 2.9 muestra un resumen de la cobertura de las diversas propuestas existentes sobre los elementos a analizar en la RQ1. En ella podemos destacar que:

- Sólo UEML permite abordar el problema utilizando diversos BPMLs, en el resto, todos los participantes de la CdS deben compartir el mismo lenguaje de modelado.
- La visión multi organizacional y la asignación de tareas a los diversos participantes es el elemento mejor cubierto en general por la mayoría de ellas.
- Más del 80 % de las propuestas no permiten establecer de forma explícita qué elementos del modelo de proceso individual son públicos y cuales privados, algo que se debe realizar de manera implícita.
- La consistencia entre la vista individual y la colaborativa es uno de los elementos menos cubierto por las propuestas existentes, algo que como veremos tiene mucho que ver con el enfoque de modelado.
- Ninguna de las propuestas soporta que los diversos participantes vean la colaboración de manera distinta.

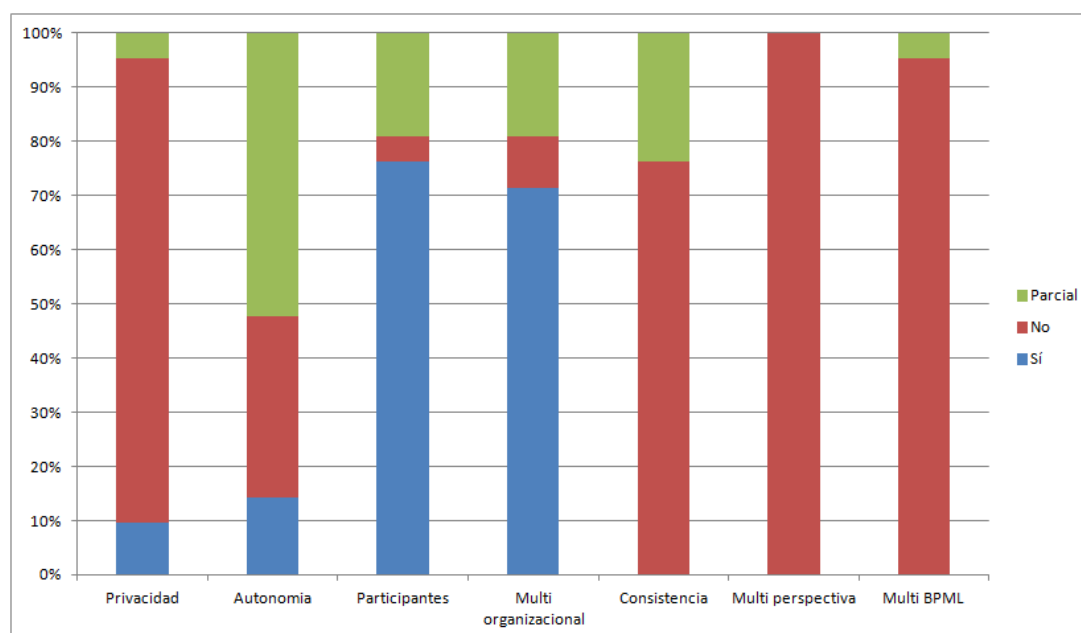


Figura 2.9: Resumen de las características analizadas en la RQ1

Por último, la figura 2.10 analiza el enfoque de modelado, en el que vemos que salvo EURL, el resto parte del diseño de la colaboración llevando a cabo un modelo Top-down, que parte de la representación de los procesos globales de la CdS, para luego detallar el comportamiento interno de cada uno. Este enfoque contradice la hipótesis de los principales modelos de referencia que asumen que para lograr una gestión de procesos inter organizacional es preciso que cada participante de la CdS gestione sus procesos de forma interna [Naslund y Williamson, 2010].

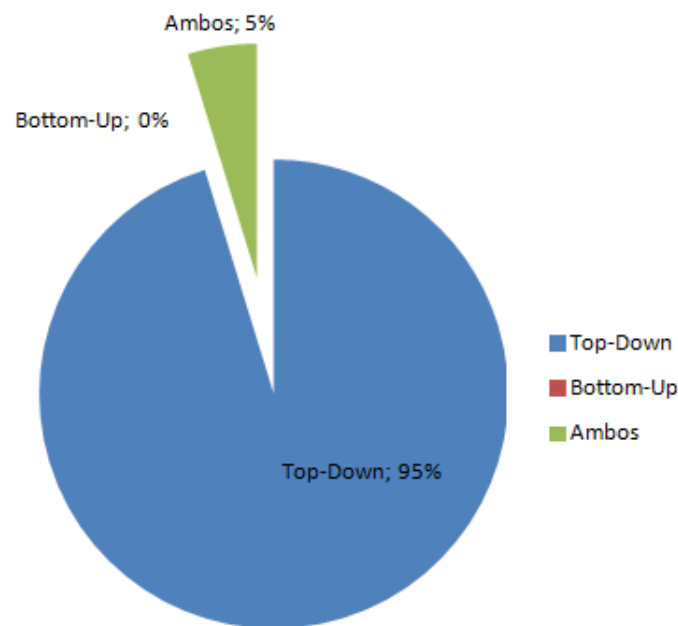


Figura 2.10: Enfoque de modelado de las propuestas existentes

Esto reafirma uno de los doce malentendidos típicos en SCM identificados por Ivanov, que las CdS surgen desde cero [Ivanov y Sokolov, 2010]. La CdS no emergen de la nada, las empresas se relacionan y crean estas estructuras virtuales con el objetivo de mejorar su operativa y aumentar el valor ofrecido a sus clientes, siendo clave la identificación de los procesos de negocio, para su posterior análisis y mejora global. Por ello creemos que este enfoque top-down puede dificultar la adopción en la práctica de estas propuestas, ya que de un modo u otro las organizaciones deben tener ya definidos y establecidos sus propios procesos internos.

Si al enfoque top-down le unimos el hecho de que sólo una propuesta plantee la posibilidad de que cada organización pueda escoger el lenguaje para modelar sus procesos de negocio internos, tenemos un marco muy poco flexible, en el que todas las organizaciones que forman parte de una o varias cadenas de valor, deben ponerse de acuerdo en utilizar un único lenguaje para representar, tanto su operativa interna, como sus relaciones modelando los procesos globales a toda la cadena.

Seguramente como consecuencia de lo anterior, más de la mitad de las propuestas no ha sido validadas, siendo el desarrollo de casos empíricos uno de los trabajos futuros más repetidos, de cara a analizar no sólo la viabilidad de las iniciativas sino, especialmente, la utilidad de tener modelos de procesos a nivel inter organizacional. Es por ello por lo que algunos autores creen que estos modelos están algo alejados de la realidad y que deben evolucionar para poder ser utilizados en la práctica [Grubic y Fan, 2010].

Por último nos gustaría destacar que las propuestas no están relacionadas y en la mayoría de las ocasiones no son analizadas como trabajos relacionados. A diferencia de lo que pasa por ejemplo en el campo de los lenguajes de modelado de procesos software, dominio conocido y bien definido, en el que existen multitud de propuestas [García-Borgoñon *et al.*, 2014] pero con fuertes relaciones, en las que unas pretenden ampliar conceptos o campos no cubiertos por otras, en el dominio del modelado de procesos de la CdS aunque se pretende resolver el mismo problema es como si los autores no fueran conscientes de los trabajos ya realizados. Ésta es una de las contribuciones de este estudio, que justificamos por la ausencia de una definición comúnmente aceptada del problema del modelado de procesos de la CdS, lo que CdS o SCM son y lo que el modelado de sus procesos debería ser.

2.3.2. RQ2: ¿Qué problemas de la CdS se han intentado resolver?

En segundo lugar nos preguntábamos qué tipo de problemática o necesidades en el ámbito de la CdS o de la SCM se han intentado resolver, algo que podemos ver para cada iniciativa en la tabla 2.10. En ella podemos agrupar los problemas en tres bloques: las que pretenden definir y representar procesos en CdS, las que buscan modelar las relaciones entre organizaciones y las que persiguen dar soporte a la toma de decisiones para la reingeniería de los procesos.

En relación a las diversas taxonomías de problemas de la SCM encontradas en la literatura, todas las iniciativas son de carácter general, es decir, pretenden representar procesos sin abordar de forma concreta la planificación, o la mejora de inventario o del transporte. Por ello vemos que el enfoque de relaciones entre los participantes de la CdS obliga a aumentar el nivel de abstracción y no centrarse en optimizar una de las operaciones concretas de una CdS, reafirmando la suposición de que SCM es un concepto más asociado a relaciones entre empresas que al dominio de la logística.

Por último, aunque todos los estudios pretenden de una u otra forma crear modelos de procesos de negocio entre empresas, hemos evidenciado que no existe un término adoptado por la comunidad para este problema. La tabla 2.11 muestra como los diferentes autores nombran a este concepto tanto en español como el término inglés. La falta de una definición comúnmente aceptada de SCM, lo que debería ser, y de los criterios para poder establecer la utilidad y la conveniencia de un modelo de procesos inter organizacional, dificulta sin duda la búsqueda de trabajos relacionados y evidencia que las diversas propuestas encontradas no tengan referencias entre ellas.

Problemas de la CdS	Referencias
Reingeniería de procesos: aplicación del modelo junto con herramientas de simulación	[Petri, 1962],[Dong y Chen, 2001],[CUI y Liu, 2005],[Zhang <i>et al.</i> , 2009],[Scheer <i>et al.</i> , 2005], [Vernadat, 2002], [OMG, 2004], [Zhang <i>et al.</i> , 2010], [Zhang, 2010], [Li <i>et al.</i> , 2008], [Li <i>et al.</i> , 2007]
Monitorización de la CdS: generación de acuerdos a nivel de procesos estratégicos y construcción de CdS ágiles a partir de los procesos	[Mayer <i>et al.</i> , 1995],[Zhou y Rong, 2010], [Eshlaghy <i>et al.</i> , 2009], [Xia <i>et al.</i> , 2006], [Li <i>et al.</i> , 2009]
Ejecución de procesos de la CdS con herramientas informáticas	[Dubray <i>et al.</i> , 2006]
Modelado conceptual y homogéneo de procesos de la CdS con un enfoque inter organizacional	[Chandra, 2008], [Chandra, 2007], [Jakjoud <i>et al.</i> , 2014]
Modelar el valor de una CdS para realizar un análisis estratégico	[Daaboul <i>et al.</i> , 2014]
Representar la trazabilidad de las actividades llevadas a cabo en la CdS	[OMG, 2015b],[Bechini <i>et al.</i> , 2008],[Kim y Rogers, 2005],[Huhns <i>et al.</i> , 2002],
Poder formar CdS en el dominio específico IT (software)	[Vo <i>et al.</i> , 2015]

Tabla 2.10: Principales problemas de la CdS que han intentado resolver las propuestas existentes

Concepto	Término en inglés	Referencias
Ingeniería de procesos de negocio entre empresas	Cross-enterprise Business Process Engineering	[Roser <i>et al.</i> , 2006]
Modelado de flujos de trabajo de la CdS	Supply Chain Workflow Modeling	[Chandra, 2008], [Chandra, 2007]
Cooperación entre múltiples empresas	Multi-enterprise Cooperation	[Hongxiu <i>et al.</i> , 2007]
Especificación de la CdS	Supply Chain Specification	[Zhang <i>et al.</i> , 2010], [Zhang, 2010], [Li <i>et al.</i> , 2008], [Li <i>et al.</i> , 2007]
Modelar la CdS	Supply Chain Modelling	[Chatfield <i>et al.</i> , 2009]
Colaboraciones de negocio entre múltiples partes	Multi-party Business Collaborations	[Hofreiter, 2009]
Modelado de procesos de negocio inter organizacionales	inter organizational Business Process Modelling	[Bouchbout y Alimazighi, 2011]
Modelo de conocimiento de la CdS	Supply Chain Knowledge Model	[Tounsi <i>et al.</i> , 2012]
Análisis estratégico de redes de valor	Strategic Analysis of Value networks	[Daaboul <i>et al.</i> , 2014]

Tabla 2.11: Términos por los que se conoce al modelado de procesos de negocio de la CdS

2.3.3. RQ3: ¿Cuáles son las limitaciones de la investigación actual?

Por último queríamos conocer cuáles eran las limitaciones en los trabajos existentes de cara a identificar posibles líneas de futuro en la materia. La tabla 2.12 muestra las líneas de trabajo a desarrollar que han sido enunciadas en cada uno de los estudios incluidos. Dejando a un lado aquellas que se centran en mejoras concretas sobre su propuesta, generalmente del lado de nuevos desarrollos o funcionalidades de las herramientas, destacamos las siguientes necesidades:

- Llevar a cabo más estudios empíricos para validar las iniciativas y evaluar los métodos y enfoques.
- Mejorar la integración del modelado de este tipo de procesos con las técnicas de simulación existentes.
- Dar soporte al enfoque bottom-up para poder reutilizar los modelos de procesos que pudieran tener ya las organizaciones.

Trabajo Futuro	Referencias
Desarrollar las transformaciones para pasar de los modelos de alto nivel creados a los sistemas informáticos propuestos	[Roser <i>et al.</i> , 2006]
Mejorar la implementación de ontologías sobre sistemas de información, mediante el desarrollo de algoritmos de búsqueda o capas de inferencia en las bases de datos para trasladar consultas entre sistemas	[Chandra, 2008], [Chandra, 2007]
Extender el alcance del modelado de un proceso de negocio, integrando dicho modelo con el análisis de requisitos, el diseño y el desarrollo de los sistemas de información asociados	[Hongxiu <i>et al.</i> , 2007]
Desarrollar una evaluación formal de la usabilidad de la herramienta. Añadir más funcionalidades para facilitar su adopción. Ampliar la base empírica del uso de EML.	[Zhang <i>et al.</i> , 2010], [Zhang, 2010], [Li <i>et al.</i> , 2008], [Li <i>et al.</i> , 2007]
Terminar de definir la terminología general de forma que el DRM (data reference model) esté separado del BPMR (business process reference model).	[Choi <i>et al.</i> , 2008]
Desarrollar un supply chain test-bed, compuesto por un conjunto de CdS de referencia, que facilite la comparativa y el análisis de la eficiencia entre diversas CdS, modelos y metodologías	[Chatfield <i>et al.</i> , 2009]
Implementar la propuesta para automatizar las transformaciones en las colaboraciones multi-empresa	[Hofreiter, 2009]
Desarrollar un caso de estudio para validar la propuesta. Integrar un enfoque de abajo a arriba (bottom-up) para poder reutilizar procesos o sistemas ya existentes antes de crear el modelo	[Bouchbout y Alimazighi, 2011]
Integrar la propuesta con técnicas de simulación. Añadir nuevos indicadores al caso de estudio. Integrar nuevos protocolos de interacción en el modelo. Añadir optimizadores en la capa de toma de decisiones (sistema de monitorización).	[Tounsi <i>et al.</i> , 2012]
Validar la propuesta a partir de casos de estudios con procesos y metodologías existentes así como en proyectos del mundo real	[Jakjoud <i>et al.</i> , 2014]
Reducir los problemas para facilitar el intercambio de datos entre las organizaciones. Analizar los riesgos en la evaluación de la eficiencia de una CdS. Automatizar los modelos de simulación incluyendo los parámetros en una base de datos.	[Daaboul <i>et al.</i> , 2014]

Tabla 2.12: Líneas de trabajo futuro identificadas en las propuestas

2.3.4. Limitaciones del estudio

Aunque hemos intentado seguir las mejores guías y prácticas propuestas para llevar a cabo una revisión sistemática de la literatura, somos conscientes de que el estudio contiene una serie de limitaciones que pasamos a detallar en esta sección.

1. En la estrategia de búsqueda

En primer lugar debemos reconocer que el número de motores de búsqueda incluidos limita los resultados obtenidos. Dado que el campo de aplicación podía ser muy diverso, desde el ámbito de la ingeniería dirigida por modelos, los procesos de negocio, la logística o la producción industrial, en lugar de identificar conferencias y revistas específicas nos decantamos por realizar búsquedas en las principales bases de datos, en particular: Google scholar, Science direct, Springerlink, Scopus, IEEEExplore y ACM Digital Library. Aun pudiendo suponer que son una muestra muy representativa de las bases de datos de publicaciones existentes, es evidente que ampliar la búsqueda con otras bases de datos podría darnos más posibles estudios candidatos, siendo por tanto una limitación en nuestro trabajo. Un posible trabajo futuro podría ser identificar qué motores de búsqueda son los que ofrecen mejores resultados para el ámbito de nuestra investigación.

Por otro lado, en cada motor de búsqueda hemos aplicado las palabras clave sobre los diversos campos permitidos, con el objetivo de obtener un número representativo de resultados sin descartar potenciales estudios en las primeras fases de nuestro análisis. Modificar o extender las palabras clave o los campos de búsqueda podría alterar de nuevo los resultados, si bien creemos que la fase de Pre-búsqueda nos ha ayudado a identificar las palabras clave y minimizar por tanto esta limitación.

2. En la selección de los estudios

Otra actividad que ha podido limitar los resultados obtenidos es la relacionada con los criterios establecidos para incluir o no los potenciales trabajos en nuestra revisión. En primer lugar, hemos decidido incluir sólo trabajo publicado en inglés, por lo que cualquier propuesta en otro idioma ha quedado excluida. Tras el análisis desarrollado, creemos que esta limitación es menor y que ningún trabajo ha quedado fuera por este motivo. También la falta de acceso al texto completo ha sido un motivo para excluir un trabajo, algo que nos ha ocurrido con 2 de las cerca de 200 potenciales iniciativas. En todo caso estos trabajos en base a su título, abstract y palabras clave estaban valorados como No y Parcialmente por los dos investigadores por lo que creemos que aun accediendo a su texto completo hubieran quedado fuera de este trabajo. Por otro lado también hemos decidido no incluir aquellas iniciativas que están más orientadas a crear modelos de CdS para su simulación, usando diversas técnicas, siendo la basada en sistemas multiagente la más frecuente atendiendo a nuestras observaciones. En este caso, cabe la posibilidad de que con ese enfoque hubiera alguna propuesta que, además de servir para su simulación, fuera de utilidad para crear un modelo de procesos y además propusiera un lenguaje novedoso. En este

caso como posible trabajo futuro quedaría el realizar una búsqueda más orientada a modelado y simulación de CdS y analizar si existen propuestas que, más allá de utilizar diversas técnicas de simulación, propongan nuevos lenguajes para abordar su modelado.

3. En el aseguramiento de la calidad

Para evaluar la calidad de los estudios incluidos definimos una serie de preguntas que nos permitiera analizar si con los mismos teníamos suficiente información como para responder a las preguntas de investigación que nos planteábamos. Para cada pregunta, hemos evaluado con un Sí, Parcialmente o No, siendo conscientes de que existe un cierto grado de subjetividad en la valoración que hemos intentado resolver mediante una puntuación individual y una reunión de consenso. A la vista de esta evaluación, hemos visto un límite en cuanto a la información relacionada con la validación de las propuestas (QA4), que hemos intentado minimizar buscando de forma iterativa nuevos trabajos que citaran a los anteriores o nuevas publicaciones de los mismos autores, sin en muchas ocasiones hallar evidencias para cubrir esta limitación. También en cuanto a las posibles líneas de trabajo futuras (QA5) existe falta de información, que hemos intentado paliar con el análisis y la retrospectiva con otros trabajos relacionados que identificaran estas líneas de trabajo. Como posible mejora podríamos definir un mayor número de QAs asociadas a cada RQ, de cara a determinar de forma más realista la calidad de los trabajos.

4. En la evaluación de la información

Por último, el propio proceso seguido para evaluar los diversos estudios supone en sí una limitación. En primer lugar, debido a que la primera fase de inclusión se realiza sobre el título, resumen y palabras clave. Extender esa primera actividad de filtrado sobre el texto completo podría darnos algún nuevo estudio en el que su contenido no hubiera sido bien resumido y expresado en los campos anteriores. Por otro lado, debemos reconocer que el análisis y evaluación de los estudios ha sido llevado a cabo por personas que, atendiendo a sus conocimientos en la materia, han ido asignando las puntuaciones y la información asociada en el esquema predefinido. A pesar de que han existido reuniones de consenso para unificar puntos de vista, el sesgo de autor y el carácter subjetivo de las valoraciones constituye una posible limitación de nuestro trabajo, que podría verse reducida en el caso de haber podido contar con un mayor número de investigadores para analizar cada estudio.

2.4. Oportunidades para la futura investigación

A la vista del análisis realizado, existen un conjunto de oportunidades y problemas sin resolver que constituyen líneas de trabajo futuro en la temática y que en esta sección resumimos. El principal reto sigue siendo el ser capaces de tener una visión holística de los procesos de la CdS [Arshinder *et al.*, 2011], de forma que las decisiones se tomen conjuntamente. Para poder alcanzar este reto, proponemos un conjunto de líneas que hemos agrupado en tres bloques: las que hacen referencia a conceptos teóricos, las que se centran en aspectos empíricos y las que definen la relación con otras áreas de investigación.

La primera línea se centraría en establecer las bases teóricas que permitan obtener una definición comúnmente aceptada del problema. En este bloque tendríamos que acotar qué es una CdS, qué entendemos por SCM, qué beneficios se podrían obtener al mejorar la integración de los procesos de negocio con un enfoque inter organizacional, cuál es el término que mejor define este problema, qué posibles enfoques existen para solucionarlo y qué indicadores, a priori, se podrían establecer para identificar si la solución es o no un éxito. Asumiendo que una de las propuestas sería trabajar en el modelado de los procesos de negocio inter organizacionales, como paso previo a su posterior análisis, simulación, optimización, promulgación o ejecución, se podría avanzar en acotar y caracterizar un proceso inter organizacional así como en los requisitos que debería cumplir un lenguaje para modelarlos, analizando el grado de cobertura de las diversas propuestas existentes frente a estas necesidades.

La segunda línea estaría ligada al análisis cuantitativo, empírico y práctico de las iniciativas. Dentro de este grupo incluiríamos el desarrollo de casos empíricos que analicen la viabilidad de llevar las diversas propuestas la práctica, las barreras existentes y beneficios obtenidos, la mejora de las herramientas que permitan la adopción de las iniciativas por parte de la industria, así como el soporte a la diversidad de lenguajes y herramientas permitiendo la reutilización de modelos existentes con un enfoque bottom-up. Manteniendo la hipótesis de que para poder llevar a la práctica de forma exitosa una gestión de los procesos de negocio entre las empresas, primero es preciso que cada organización gestione sus procesos [Naslund y Williamson, 2010], con la diversidad de combinaciones posibles entre empresas, lenguajes utilizados y CdS en las que participa, ser capaces de que cada empresa utilice un lenguaje y existan mecanismos para transformar esos modelos en un lenguaje común sigue siendo un gran reto pendiente.

Por último la tercera línea plantea desarrollar un enfoque multi-disciplinar, definiendo el problema junto con los interfaces o potenciales relaciones existentes con otros ámbitos de la investigación. Para disponer de modelos de procesos de negocio inter organizacionales, creemos que es necesario reutilizar los resultados existentes en otros dominios, como el de la ingeniería de procesos de negocio, o el de las técnicas de simulación, o los sistemas de ayuda a la toma de decisiones, o la ingeniería de sistemas o incluso las arquitecturas para la integración de sistemas informáticos. Es preciso por tanto establecer un marco que analice qué entradas y salidas se pueden establecerse entre otras áreas y el campo del modelado de procesos de negocio de la CdS.

A la vista de otros trabajos de revisión realizados las líneas futuras identificadas se alinean con muchas de las ya propuestas por los diversos autores, como vemos en la tabla 2.13, en la que en el primer bloque analizamos revisiones en el ámbito de la CdS o de SCM, en el segundo trabajos sobre lenguajes de modelado de procesos de negocio y en el tercero incluimos las revisiones sobre modelado de la CdS.

Año	Línea Futura	Referencia
2005	Colaborar para descentralizar la toma de decisiones en la CdS así como conseguir una integración de las diversas operaciones entre las organizaciones	[Stadtler, 2005]
2009	Abordar la resolución de problemas con un enfoque de CdS, en lugar de utilizar la perspectiva individual de una organización	[Stock, 2009]
2010	Mejorar en el modelado de la CdS, evitar tener el enfoque de que la CdS surge de la nada	[Ivanov y Sokolov, 2010]
2011	Tener un enfoque holístico en la CdS	[Arshinder <i>et al.</i> , 2011]
2012	Desarrollar estudios empíricos	[Noordin <i>et al.</i> , 2012]
2013	Desarrollar una base teórica que permita evaluar estudios cuantitativos	[Kim <i>et al.</i> , 2013]
2004	Crear un marco general que establezca lo que una técnica de modelado de procesos de negocio debe incluir para que sea exitosa	[Aguilar-Saven, 2004]
2010	Llegar a disponer de un estándar de modelado de procesos de negocio, algo que finalmente no se ha llegado a alcanzar	[Mili <i>et al.</i> , 2010]
2002	Métodos analíticos que permitan analizar y explotar las ventajas derivadas de la integración de los conceptos de la CdS	[Min y Zhou, 2002]
2010	Mejorar los conceptos en los modelos para estar más cercanos a la realidad	[Grubic y Fan, 2010]
2012	1) Modelos y herramientas multi-objetivo para mejorar la integración y coordinación en la CdS; 2) métricas que evalúen de forma práctica el rendimiento de una CdS; 3) aplicación de las nuevas tecnologías para dirigir la ejecución de las actividades y procesos de la CdS	[Badole <i>et al.</i> , 2012]

Tabla 2.13: Líneas de trabajo futuro identificadas en otras revisiones de la literatura

2.5. Conclusiones

En este capítulo se ha presentado el estudio del estado del arte de los diferentes lenguajes de modelado de procesos de la CdS existentes a partir de una revisión sistemática de la literatura. A la vista de los resultados hemos evidenciado una evolución del nivel de abstracción de los modelos de procesos de la CdS, donde lo relevante no es detallar la operativa sino entender y analizar las relaciones entre los participantes, para lo cual el paradigma basado en modelos está siendo el más utilizado. La falta de una definición comúnmente aceptada del problema y sus términos dificulta el análisis de los trabajos realizados y hace que las diversas propuestas existentes sean independientes. El enfoque top-down, en el que se parte de la descripción de los procesos globales a la CdS, y la falta de interoperabilidad entre lenguajes y modelos, obliga a ponerse de acuerdo a todas las organizaciones en el uso de un mismo lenguaje para representar tanto operativa interna como la colaboración en la CdS, lo que dificulta enormemente su adopción en la práctica, algo que se evidencia en la falta de estudios empíricos en la materia.

Como resultado de la revisión sistemática se han observado diferentes aspectos como líneas de trabajo futuras, resumidas en la capacidad de disponer de una visión holística en la que las decisiones se tomen conjuntamente, y que hemos agrupado en tres bloques:

- Establecer las bases teóricas que permitan obtener una definición comúnmente aceptada del problema.
- Desarrollar estudios empíricos para analizar cuantitativamente barreras y mejoras.
- Crear un enfoque multi-disciplinar combinando resultados de otras áreas de investigación.

Como lecciones aprendidas para nuestro trabajo, hemos evidenciado que **no existe ninguna propuesta que cubra las necesidades** que hemos descrito en el capítulo anterior, en particular lo relativo a:

- Privacidad: permita establecer qué elementos del modelo individual son visibles en la colaboración y cuáles no.
- Enfoque bottom-up: permita generar el modelo de proceso de la CdS reutilizando los modelos individuales y garantizando por tanto la consistencia entre ambos modelos.
- Multi BPML: Soporte para que cada organización pueda utilizar el BPML más adecuado.
- Multi perspectiva: Soporte para que cada organización pueda ver la colaboración de una forma distinta.

De todas las propuestas existentes, BPMN sería la más cercana a cumplir nuestras necesidades, si bien no nos permite las dos últimas características solicitadas. Por otro lado, UEMML es la propuesta más cercana a dar ese soporte multi BPML, si bien no hemos encontrado evidencias de que haya tenido la adopción inicialmente esperada.

Manteniendo la hipótesis de que para poder llevar a la práctica de forma exitosa una gestión de los procesos de negocio entre las empresas, primero es preciso que cada organización gestione sus procesos [Naslund y Williamson, 2010], de cara a nuestro trabajo de investigación a partir de esta revisión hemos evidenciado la necesidad de poder abordar el modelado de procesos de negocio de la CdS con un nuevo enfoque, bottom-up, en el que se dé soporte a una diversidad de lenguajes y herramientas y se reutilicen los modelos existentes, **no habiendo ninguna propuesta hasta la fecha** con estas propiedades.

Capítulo 3

Planteamiento del Problema

En el capítulo anterior hemos analizado el estado del arte en cuanto a los lenguajes de modelado de procesos de negocio de la CdS, en el que hemos visto que existen multitud de propuestas, independientes entre sí, que pretenden dar respuesta a la necesidad creciente de tener un entorno basado en modelos que represente las interacciones y dependencias que existen entre los procesos que forman parte de una cadena de valor. Este análisis nos ha permitido determinar las necesidades existentes que pretendemos cubrir dentro de esta tesis y que, junto con la estructura de la solución, componen el principal objetivo de este capítulo que se estructura como sigue. En la primera sección se establece el contexto del problema, a través de un conjunto de aspectos que consideramos relevantes para determinar el problema a resolver, que se detallan a continuación. En la tercera sección se enumeran los objetivos a alcanzar, precediendo a las principales influencias que hemos tenido a la hora de plantear la solución, cuyo planteamiento general se expone en la quinta sección. Finalmente, la última sección resume las principales conclusiones de este capítulo.

3.1. Aspectos relevantes que determinan el problema a resolver

En esta sección vamos a presentar un conjunto de aspectos que consideramos relevantes para determinar el problema a resolver y que vienen derivados tanto del contexto de investigación, presentado en la introducción, como de las conclusiones obtenidas en el estudio del estado del arte expuesto en el capítulo 2. En ellas hemos identificado:

- La necesidad actual de las organizaciones de ser capaces de definir y mejorar sus procesos con el objetivo de aumentar el valor ofrecido al cliente.
- La dificultad a la hora de poder implementar las mejoras sin antes analizar las consecuencias en un entorno basado en modelos dinámicos.
- La gran diversidad existente en cuanto a lenguajes de modelado de procesos de negocio con potencial aplicación en la CdS.

3.1.1. Las organizaciones necesitan definir y mejorar los procesos de su cadena de valor

Ya hemos visto en la introducción que en la actualidad la competencia en los mercados no está entre las compañías individuales, sino entre las diversas cadenas de valor formadas por conjuntos de ellas que dan satisfacción al cliente final. Lo que se persigue no es tanto mejorar la operativa interna e individual de cada organización, sino la eficiencia global a partir de las interacciones entre las actividades y procesos desarrollados por cada uno de los participantes de la cadena.

Por otro lado una misma organización puede formar parte de un gran número de cadenas de suministro, jugando un rol diferente en cada una. Pensemos por ejemplo en una empresa de transporte, que distribuye productos de muy diversas compañías, siendo el principal agente para algunas, mientras que en otras pasa a ser un participante secundario sólo en casos excepcionales. La compañía se especializa en garantizar entregas en plazo y ajustar sus costes, para lo que dispone de unos procesos internos orientados a la planificación del transporte y a la gestión de las incidencias. Su operativa interna y sus decisiones están definidas y cuantificadas en términos operativos como empresa.

En este contexto, crear modelos de las cadenas de valor conlleva una complejidad mayor que el modelado de los procesos de negocio de cada participante, dado que cada uno de los miembros querrá preservar:

- Su privacidad en sus procesos internos y en la información que gestiona.
- Su autonomía a la hora de tomar las decisiones, maximizando su beneficio mientras se cumplan los compromisos adquiridos con el resto de la CdS.

Siguiendo con el ejemplo anterior, para esa organización de transporte, ser consciente del papel que ocupa en cada una de las cadenas de valor en las que participa y conocer cómo sus decisiones implican en la mejora de toda la red es necesario, si bien se antoja como una difícil tarea. Sólo en CdS poco dinámicas, en las que pueda existir una figura principal sobre el resto (por ejemplo en una CdS de automoción en la que un gran fabricante orquesta a sus proveedores de primer y segundo nivel), sobre las que existan procesos pre-establecidos con indicadores globales e individuales, se dispone de verdaderos modelos extremo a extremo desde el aprovisionamiento hasta la entrega al cliente.

3.1.2. La mejora de los procesos de la CdS requiere de entornos basados en modelos dinámicos, flexibles y adaptables

En el contexto de una organización individual, analizar las consecuencias derivadas de una potencial mejora en sus procesos es relativamente factible, tanto a nivel analítico como en la realidad, dado que el número de factores o variables sobre los que se condiciona el éxito de la mejora son conocidos y controlados por una misma entidad. Sin embargo, esto no ocurre en el contexto de la CdS, donde el análisis de las consecuencias de un cambio puede venir determinado por un conjunto de factores elevado y, en ocasiones, indeterminado, derivado de la estructura de la CdS en la que los participantes mantienen su autonomía en la toma de decisiones. Además los procesos en una CdS tienen una característica de temporalidad importante, es decir, el ciclo de vida de un proceso puede durar días, semanas o incluso meses, por lo que han podido cambiar las condiciones del contexto a la hora de poder valorar en el mundo real si un cambio producido ha tenido o no el impacto esperado.

Es por ello que los entornos basados en modelos, en los que se pueden representar las diversas variables que afectan al proceso global y analizar, utilizando diversas técnicas de otras disciplinas como la simulación o la optimización basada en investigación de operaciones, de forma que se pueden cuantificar al menos a nivel teórico las ventajas o beneficios que se podrían obtener en una mejora. Es evidente que pueden existir diferencias en el momento de la implementación, pero disponer de entornos controlados, en los que se pueda experimentar y analizar las consecuencias antes de su implantación en la realidad se antoja como algo necesario.

Pero además las CdS se caracterizan por su heterogeneidad y dinamismo [Sadiq *et al.*, 2005], lo que obliga a que los modelos sean:

- *Dinámicos*, que puedan ser modificados.
- *Flexibles*, cuyas instancias puedan evolucionar en tiempo de ejecución, añadiendo y replanificando actividades, atendiendo a un marco definido en tiempo de diseño.
- *Adaptables*, pudiendo generar instancias de modelos específicos que incluyen particularidades o circunstancias excepcionales que pueden o no haber sido previstas antes del comienzo de la ejecución.

Sobre el 60 % de la actividad de un trabajador en la industria está relacionado con trabajo no predefinido e, incluso, imprevisible [Swenson, 2011]. En ocasiones podríamos estar haciendo referencia de los procesos basados en el conocimiento (*Knowledge Intensive Processes*, KIP) [Isik *et al.*, 2013], en los que no se puede establecer a priori una lógica secuencia de actuaciones sino que será en tiempo de ejecución cuando una persona, basándose en la información disponible y en su conocimiento, ordene y ejecute las tareas pertinentes.

Por ello es necesario poder modificar instancias de modelos de procesos en tiempo de ejecución como requisito fundamental para poder llevar a cabo de forma efectiva un modelado para soportar este dinamismo y que, incluyendo esta flexibilidad y adaptación de los modelos de procesos, nos permita tomar decisiones en colaboración a partir del modelado de los procesos de negocio con un enfoque inter organizacional. Para generar estos modelos de proceso existen una gran variedad de lenguajes, que exponemos a continuación.

3.1.3. Existen muchos lenguajes de modelado de procesos de negocio y de la CdS

Una vez hemos visto la necesidad de las organizaciones de gestionar sus procesos con un enfoque de CdS, y la utilidad de poder generar modelos flexibles y adaptables de esos procesos, llegamos al punto en el que deben seleccionar un lenguaje para poder crear esos modelos de los procesos. En la introducción vimos la gran cantidad de lenguajes existentes para modelar procesos de negocio y en la revisión del estado del arte encontramos más de diez propuestas específicas para los procesos de la CdS.

Así que para una organización, la selección del lenguaje más apropiado para crear un modelo de sus procesos no es una tarea sencilla. Además, una vez que han realizado la apuesta por uno, se crea un vínculo entre el modelo y el lenguaje utilizado, por lo que migrar de lenguaje de modelado no es una tarea libre de riesgos, por inconsistencias en los modelos, ni esfuerzo, por el trabajo a desarrollar de nuevo. Si a esto le unimos el hecho de que, una vez que una organización gestiona sus procesos, lo que pretendemos es crear un modelo global que integre todos los procesos de los diversos participantes en la CdS, ante la falta de interoperabilidad entre los modelos creados en base a diversos lenguajes, se hace preciso que todas las organizaciones de una CdS apuesten por el mismo lenguaje o bien migren sus modelos a dicho lenguaje.

Pero, aunque sea factible que todas las organizaciones de una CdS creen sus modelos de procesos con el mismo lenguaje, es evidente que resta flexibilidad a la solución por cuanto que es habitual que una empresa participe de varias CdS, por lo que con casi total seguridad se verá obligada a ir migrando sus procesos a los diversos lenguajes de cada CdS. Para resolver este problema, como influencias a nuestro trabajo veremos que existe una propuesta [García-Borgoñon, 2016] que pretende resolver este problema mediante la creación de un marco de referencia que facilita la interoperabilidad de diversos modelos de procesos creados con lenguajes diferentes, y que constituirá una base sobre la que construiremos nuestra solución.

3.2. Planteamiento del problema a resolver

Seguidamente pasamos a detallar el planteamiento del problema a resolver. En la primera sección se expone la necesidad detectada y las ideas novedosas en cuanto a su resolución. Seguidamente se detalla el problema mediante una serie de casos de uso. Por último, se enuncia el problema a nivel teórico.

3.2.1. Necesidad a resolver

A la vista del contexto anteriormente presentado, podemos destacar que las organizaciones necesitan disponer de modelos flexibles y adaptables de los procesos de sus CdS, que incluyan la suma e interacción de los procesos individuales y autónomos de cada participante, como marco de análisis y mejora antes de su implantación en la realidad. También hemos evidenciado la dificultad a la hora de elegir un lenguaje de modelado de procesos, algo que se enfatiza si además queremos que sea válido y satisfaga a todos los participantes de la CdS.

La necesidad de fondo que perseguimos es poder disponer de una solución que permita **crear modelos de procesos colaborativos**, o dicho de otra forma, se trata de modelar coreografías de procesos que representan las interacciones que se hacen entre cajas negras de procesos individuales que son llevados a cabo por organizaciones distintas.

Este problema ha sido cubierto en la literatura, pero siempre con un enfoque *top-down*. Cabe destacar algunas iniciativas desarrolladas desde los ámbitos empresariales en diversos dominios de aplicación, como GS1 [Hübner, 2008] (anteriormente RosettaNet [Damodaran, 2004]) para la logística, OTA [Dogac *et al.*, 2004] para turismo, HL7 [Dolin *et al.*, 2006] para salud o SWIFT [Baker y Byler, 1983] para servicios financieros entre otros, que han intentado establecer estándares para generar reglas por defecto a estas coreografías de procesos.

Si bien estos estándares han tenido su importancia en sus dominios de aplicación particulares, adolecen de la flexibilidad necesaria para poder definir nuevos tipos de colaboraciones que faciliten la cooperación entre empresas en el mercado dinámico actual.

Es aquí donde nuestro trabajo aporta una idea novedosa, en cuanto a la capacidad de abordar el problema con otra perspectiva, *bottom-up*, reutilizando los modelos de los procesos que cada organización pueda haber creado con sus lenguajes de modelado, añadiendo a esa vista individual de la colaboración las restricciones a la coreografía de los procesos con el resto de participantes de la CdS.

Esta nueva forma de abordar el problema del modelado de procesos de negocio colaborativos, la llevaremos a cabo en nuestro trabajo haciendo uso del paradigma guiado por modelos, base que ha sustentado nuestra hipótesis de investigación, si bien en el futuro podría ser acometida con otras tecnologías.

Los principales beneficios derivados de este enfoque a la hora de resolver el problema son:

- Al reutilizar los modelos de procesos individuales se reduce el riesgo de tener incoherencias entre el modelo individual y el colaborativo.
- Se da soporte a que cada organización utilice el lenguaje de modelado de procesos de negocio más idóneo para sus necesidades.
- Se permite que una misma colaboración pueda ser vista de manera diferente por diversos miembros de la CdS. Lo que para uno puede ser un proceso estructurado, para otro participante puede ser un caso dirigido por los datos y con planificación dinámica. Independizamos de esta forma a nivel conceptual cómo cada organización ve cada colaboración.
- De igual forma cada compañía decide, para cada colaboración en la que participa, cómo preservar su autonomía en la toma de decisiones y la privacidad en lo relativo a su proceso interno.
- El mercado demanda cambios constantes y el enfoque de modelado debe estar diseñado para que esos cambios se puedan introducir con agilidad. El enfoque *bottom-up* nos permite añadir fácilmente las nuevas coreografías requeridas para colaborar en una nueva CdS o para colaborar con una nueva organización en la cadena.

3.2.2. Casos de uso

De cara a facilitar la descripción del problema a resolver vamos a presentar un conjunto de casos de uso que utilizaremos a lo largo del presente trabajo como muestra de la necesidad y validación de la propuesta:

1. La creación de modelos de procesos de CdS reutilizando modelos de procesos individuales. Supongamos que un conjunto de organizaciones forman una CdS, una fabrica materias primas, otra ensambla el producto y otra lo transporta hasta el cliente. Cada una de ellas dispone de modelos de procesos individuales, utilizando distintos lenguajes de modelado, y se plantean crear un modelo de proceso global que incluya desde el aprovisionamiento de las materias primas, hasta su entrega al cliente, pasando por su fabricación. Este modelo global les permitirá tener una visión orientada al cliente y establecer los indicadores del proceso general en base a la satisfacción del cliente. En este escenario cubriremos la posibilidad de representar un proceso con un lenguaje, construir a partir del mismo una visión integrando la colaboración con otros participantes para volver a tener un modelo de dicha colaboración en el lenguaje original, es decir, integrar con un mismo enfoque de modelado la vista individual y colaborativa de un proceso.
2. Pensemos que en el caso anterior, en el que ya teníamos la CdS formada y su proceso general modelado, deseamos incorporar nuevas organizaciones, como pueden ser almacenes intermedios, más compañías de transporte o de materias primas. Cada una de ellas tendrá sus procesos modelados, y posiblemente con diversos lenguajes. En este caso lo que buscamos es analizar la flexibilidad de la solución para incorporar el dinamismo natural de

las CdS en la actualidad, es decir, la incorporación de nuevos procesos a la colaboración sin por ello tener que empezar de cero o condicionar el proceso colaborativo ya disponible.

3. Centrémonos en una empresa individual que tiene sus procesos modelados con un lenguaje, y que participa en diversas CdS en las que cumple un rol diferente. En este caso lo que buscamos es analizar cómo la propuesta da soporte a la integración entre la gestión de procesos individual y la global con un enfoque de CdS. Con este escenario cubriremos la posibilidad de que una organización pueda participar en múltiples CdS que a su vez han podido ser modeladas atendiendo a diversos enfoques o lenguajes.

Cabe destacar que cuando estamos hablando de organizaciones distintas en una cadena de valor, bien podríamos estar en el caso de grandes organizaciones que tienen áreas funcionales en su estructura. Por ejemplo, una organización podría tener un departamento de compras, otro de producción y otro de logística, encargados desde el aprovisionamiento hasta la entrega de los productos al cliente, si bien cada uno de ellos podría tener sus procesos de negocio específicos, con sus indicadores e incluso sus modelos con diversos lenguajes. Aunque el problema sea similar al caso de varias empresas diferentes, y que la solución propuesta también pueda ser utilizada, este caso vamos a dejarlo fuera de nuestro contexto dado que en la comunidad el concepto de CdS está admitido que representa dos o más entidades independientes, entendiendo por ellas organizaciones jurídicamente distintas.

Los elementos comunes que tienen todos los casos de uso son los propios de los modelos de procesos inter organizacionales, como son:

- Existe privacidad en los procesos individuales, especialmente en lo relativo a objetivos de negocio e indicadores.
- Se mantiene la autonomía en la toma de decisiones en cada organización.
- El objetivo de cada participante es tomar las decisiones que maximizan su beneficio manteniendo el cumplimiento de los compromisos adquiridos en la CdS.

El problema a resolver se sustenta sobre dos conceptos generales de aplicación al modelado de procesos de negocio de la CdS:

- El dinamismo, entendido como su capacidad de modificación asociando las características de flexibilidad y adaptación en los modelos de procesos.
- El enfoque inter organizacional, en el que se construye el proceso general de la CdS mediante la composición de los modelos individuales que son reutilizados.

3.2.3. Marco teórico

Por último presentamos el marco teórico del problema a resolver, a través de una extensión del modelo conceptual de procesos de negocio colaborativos propuesto por Gong et al. [Gong *et al.*, 2006] para dar soporte a las necesidades de toma de decisiones en una CdS, tal y como se expone en la figura 3.1.

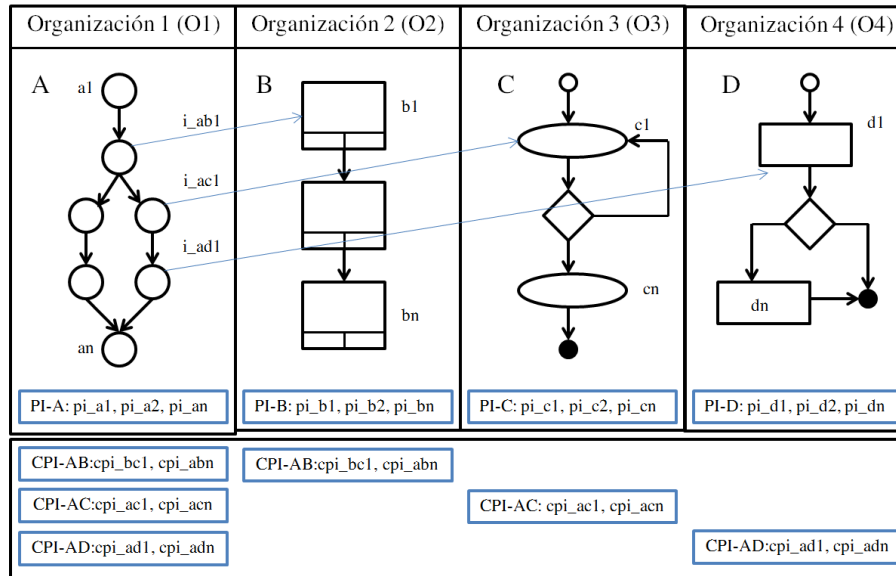


Figura 3.1: Planteamiento teórico del problema a resolver

Sea SC una CdS formada por O_1 , O_2 , O_3 y O_4 que representan diversas organizaciones. Sean A , B , C y D distintos procesos: $A = \{a_1, a_2, \dots, a_n\}$, $B = \{b_1, b_2, \dots, b_n\}$, $C = \{c_1, c_2, \dots, c_n\}$, $D = \{d_1, d_2, \dots, d_n\}$, en los que a_i o b_i o c_i o d_i representan actividades. Cada proceso de negocio ha sido modelado con un lenguaje diferente ya que en un ámbito industrial es común que diversas técnicas de modelado de procesos puedan llevarse a cabo. Cada proceso tiene su propio indicador (*Process Performance Indicator, PPI*), representado por el conjunto $PI = \{pi_1, pi_2, \dots, pi_n\}$ y alineado con los objetivos de su organización. En su cadena de valor, los procesos A , B , C y D deben ejecutarse en cooperación para satisfacer un objetivo de negocio global. Esta visión general de los procesos implica que debe existir un intercambio de información entre los procesos. En nuestro caso realizaremos una extensión de esta visión de forma que las actividades que requieran tomar una decisión puedan ser colaborativas en este modelo conceptual, de manera que demos soporte a estos escenarios:

- El proceso A toma de forma autónoma una decisión y notifica sus implicaciones al proceso B .
- El proceso A realiza una petición al proceso C antes de llevar a cabo la decisión.

- El proceso A establece una negociación con el proceso D para tomar una decisión colaborativa. Este tercer escenario de colaboración requiere la definición de un indicador conjunto (*Collaborative Process Performance Indicator*, $CPPI$), que representamos por el conjunto $CPI = \{cpi_1, cpi_2, \dots, cpi_n\}$ y que influye en el proceso de toma de decisión para alcanzar un objetivo común entre las partes.

En estos escenarios el estado de ejecución de los diversos procesos debe ser controlado o adaptado dinámicamente en función de la información intercambiada (*Process Collaboration Information*, PCI), que está representada por el conjunto $I = \{i_1, i_2, \dots, i_n\}$ y que será enviada a través de interfaces de procesos. De esta forma, el comportamiento dinámico de un proceso emerge con la misma importancia que su vista estática, dado que cualquier proceso puede ser adaptado en función de la información intercambiada en tiempo de ejecución.

Una vez enunciado el planteamiento del problema a resolver pasamos a detallar los objetivos a alcanzar en este trabajo de tesis.

3.3. Objetivos a alcanzar

Una vez que hemos presentado el contexto y problema a resolver, concretado a partir de tres casos de uso, en esta sección enumeramos los principales objetivos a alcanzar, que son los siguientes:

1. Realizar un estudio del estado del arte sobre lenguajes de modelado de procesos de la CdS. Este análisis nos permitirá conocer con mayor grado de detalle las propuestas ya realizadas, los problemas que han intentado abordar así como las líneas de trabajo abiertas en la temática.
2. Definir un lenguaje de modelado de procesos de negocio colaborativos que; 1) permita reutilizar modelos existentes y; 2) se centre en las relaciones entre los procesos garantizando la privacidad de la visión interna de cada participante así como su autonomía a la hora de tomar decisiones.
3. Definir un método para, a partir de modelos de procesos individuales descritos en diversos lenguajes, poder construir el modelo de proceso colaborativo que representa la vista inter organizacional y las interacciones entre los mismos.
4. Dar soporte a la adaptación y flexibilidad de los procesos colaborativos, por lo que éstos contarán con una parte estática y una dinámica. Por ello se hace preciso definir un lenguaje de modelado de casos colaborativos que incorpore esa capacidad de planificación dinámica sobre la estructura estática inicialmente planteada.
5. Definir un marco conceptual basado en transformaciones entre modelos que nos permita pasar los modelos de procesos de negocio colaborativos a modelos de casos colaborativos, añadiendo toda la capacidad de planificación dinámica de los últimos. De esta forma seremos capaces de mantener un enfoque de reutilización de modelos incluso a partir del modelo de proceso colaborativo.
6. Desarrollar una herramienta que dé soporte para que este marco conceptual que pueda ser utilizado en la práctica por organizaciones.
7. Validar la propuesta con un caso de estudio real.

3.4. Influencias conceptuales y tecnológicas

Antes de exponer la estructura de la solución, vamos a enunciar aquellos métodos o técnicas que han influido en la forma en la que planteamos abordar la solución al problema anteriormente expuesto. Entre ellas, se encuentran las lecciones aprendidas tras la revisión sistemática de la literatura, el paradigma basado en modelos, la gestión de casos para representar procesos dinámicos dirigidos por datos, así como la propuesta para facilitar la interoperabilidad de los modelos de procesos conformes a cualquier lenguaje.

3.4.1. Las lecciones aprendidas de la revisión sistemática de la literatura

Como se ha expuesto en el capítulo 2, el análisis de la situación actual en lo relativo a los lenguajes de modelado de procesos de negocio de la CdS llevado a cabo mediante una revisión sistemática de la literatura, nos ha permitido identificar las principales áreas de investigación futuras así como asentar algunas de las hipótesis que teníamos al comienzo de nuestro trabajo. Seguidamente pasamos a exponer aquellos aspectos más relevantes como lecciones aprendidas que influyen en la propuesta de solución planteada.

En primer lugar hemos podido evidenciar que la gestión de los procesos de negocio en el ámbito de la CdS es similar a cualquier otro dominio específico, si bien conlleva dos grandes particularidades: la importancia de tener modelos de procesos, dado que el análisis de las potenciales mejoras derivadas de un cambio son difíciles de experimentar en la realidad, y la inherente característica de dinamismo y heterogeneidad presente en dichos modelos de procesos.

En segundo lugar se ha manifestado la gran falta de estudios empíricos en la materia que analicen y demuestren la viabilidad práctica de las propuestas e identifiquen los beneficios obtenidos y las principales barreras para su adopción. En nuestra opinión la falta de reutilización de los posibles modelos de procesos previos que tuvieran las organizaciones supone en sí una gran dificultad para llevar a cabo esta tarea.

Y por último la falta de conexión y relación entre las diversas propuestas de lenguaje de modelado de procesos de negocio de la CdS, algo que evidencia que se pretenden resolver problemas específicos no siempre independientes sin que hasta la fecha encontremos soluciones que apuesten por facilitar la interoperabilidad entre las diversas propuestas planteadas.

Por todo ello tras esta revisión sistemática hemos podido asentar alguna de nuestras creencias de partida y reafirmar que nuestra propuesta, basada en un nuevo enfoque bottom-up para la creación de modelos de procesos de negocio de la CdS, en el que se da soporte a una diversidad de lenguajes mediante la reutilización de modelos existentes, puede tener cabida en este dominio específico y puede resolver alguno de los problemas evidenciados. Seguidamente se expone cómo los paradigmas basados en modelos y en servicios han influido en la propuesta de solución planteada.

3.4.2. La ingeniería del software basada en modelos

En el capítulo introductorio ya hemos analizado el contexto en el que la ingeniería del software dirigida por modelos tenía potencial aplicación en el ámbito del modelado de procesos de negocio de la CdS.

Una de las conclusiones del estudio del estado del arte ha sido evidenciar una evolución del nivel de abstracción de los modelos de procesos de la CdS, donde lo relevante no era detallar la operativa sino entender y analizar las relaciones entre los participantes, para lo cual el paradigma basado en modelos estaba siendo el más utilizado. Por ello, del paradigma basado en modelos vamos a adoptar sus dos principales pilares, el metamodelado, como elemento para aumentar la abstracción y las transformaciones entre modelos, como facilitador para aumentar la interoperabilidad entre modelos creados con distintos lenguajes.

De forma particular, la figura 3.2 muestra cómo desde MDE se aborda el modelado de las coreografías de procesos de negocio. Las instancias de procesos en ejecución (nivel M2) establecen conversaciones entre sí, atendiendo a la coreografía definida en el modelo del proceso (nivel M1), que a su vez se basan en un metamodelo de coreografías (nivel M2).

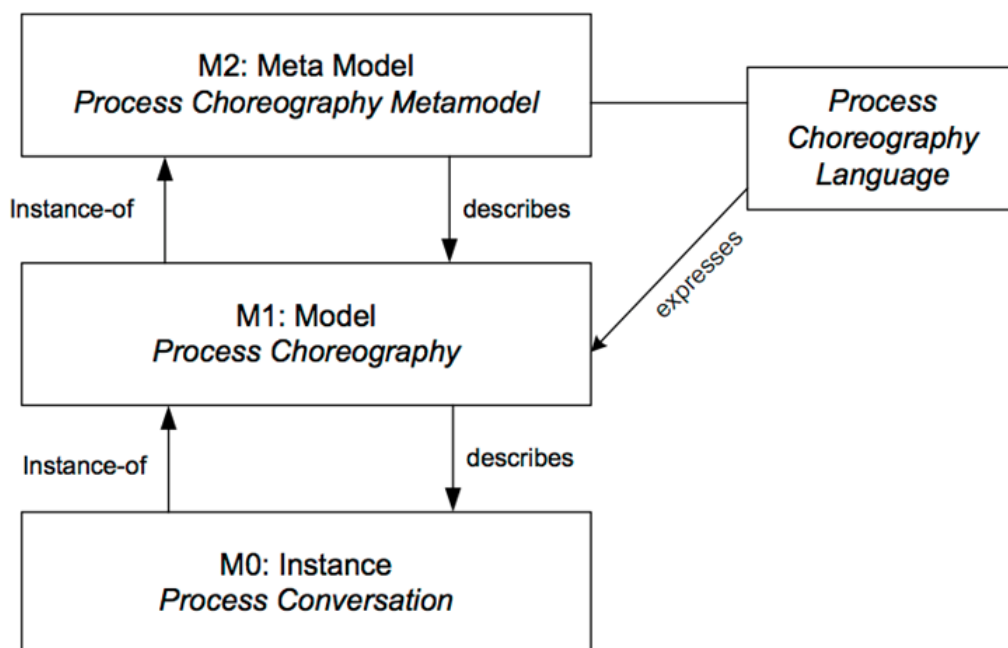


Figura 3.2: Niveles MOF de las coreografías de procesos [Weske, 2010]

En MDE, las operaciones sobre los modelos se realizan en base a transformaciones [Sendall y Kozaczynski, 2003]. Dentro del marco CBG vamos a hacer uso de las transformaciones modelo a modelo para realizar el paso del dominio de procesos colaborativos al de casos colaborativos.

El estándar *Query/View/Transformation (QVT)* se compone en realidad de tres lenguajes diferentes, con un enfoque híbrido declarativo e imperativo, en el que la parte declarativa se divide en dos niveles. De esta forma QVT establece dos lenguajes declarativos, *QVT Relations* y *QVT Core*, y uno imperativo, *QVT Operational Mappings*, *OM*.

El hecho de que QVT tenga diversos enfoques obedece a la creencia, por parte de varios autores, de que los enfoques declarativos son mejores para especificar simples transformaciones y relaciones entre los elementos de los modelos origen y destino, mientras que las propuestas imperativas nos permiten resolver problemas de transformación más complejos [Romeikat *et al.*, 2008]. De forma resumida, como se muestra en la figura 3.3:

- El lenguaje *QVT Relations* dispone de un complejo sistema de coincidencia de patrones, soporte la bidireccionalidad de las transformaciones y contiene una sintaxis gráfica.
- El lenguaje *QVT Core* es más simple, estableciendo la coincidencia de patrones únicamente para un conjunto de variables.
- *QVT Operational Mappings* es un lenguaje imperativo que proporciona extensiones de OCL [OMG, 2012] con un estilo meramente procedimental, dado que en ocasiones para la definición de transformaciones complejas no es posible el uso de un enfoque puramente declarativo [Gardner *et al.*, 2003]. A través del mecanismo *BlackBox* es posible utilizar lenguajes de programación externos con la idea de reutilizar librerías externas de cara a implementar las transformaciones. Este mecanismo, aunque existente, no es muy recomendable por las posibles inconsistencias que puedan surgir durante su uso.

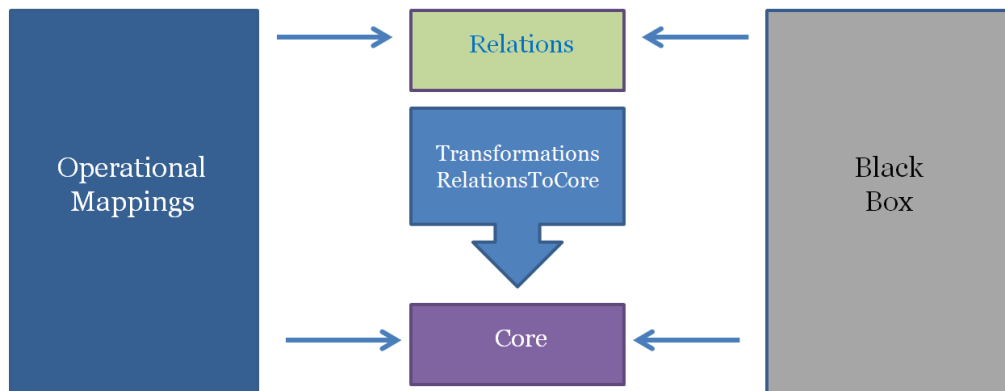


Figura 3.3: Los lenguajes de QVT: relaciones y arquitectura

QVT Relations define una sintaxis concreta textual mediante una gramática cuya especificación completa se encuentra en [OMG, 2011a]. En ella, una **transformation** establece las relaciones bidireccionales que existen entre los elementos de dos modelos. Las **top relation** son aquellas que de obligada ejecución, mientras que el resto son invocadas directamente dentro de alguna cláusula de otra relación.

Una relación puede estar condicionada por dos predicados, las cláusulas **when** y **where**. La primera especifica las pre-condiciones que deben cumplirse por todos los elementos del modelo que participan en la relación y puede requerir el cumplimiento de ciertos valores en las variables y sus dominios. La segunda especifica las condiciones que deben mantenerse en la relación. Cuando una transformación se ejecuta en la dirección de un dominio **checkonly**, implica que tan sólo se comprueba si existe un dominio válido en el modelo que satisface la relación. En un dominio **enforce**, en el caso de que no existiera (lo que implica que ese chequeo de que en ambos modelos existan los elementos que cumplen la relación falla), el modelo destino es modificado para poder satisfacer la relación. Las relaciones incluyen expresiones en las que se establecen los mapeos entre los elementos del modelo origen y los declarados dentro del dominio que, a su vez, puede incluir variables libres sin asignar o pueden estar ya con un valor como resultado de la ejecución de otras evaluaciones o relaciones.

El algoritmo 3.1 resume la sintaxis textual de QVT.

```

1 % Start your code-block
2
3 transformation ... {
4   top relation R {
5     checkonly|enforce domain a...
6     checkonly|enforce domain b...
7     when {...}
8     where {...}
9   }
10
11   top relation S ...
12   relation ...
13   relation ...
14   ...
15 }
```

Algoritmo 3.1: Versión simple de la sintaxis concreta textual de QVT Relations

QVT Relations también incluye una sintaxis gráfica para representar transformaciones, dominios y patrones, representada en la figura 3.4. En ella vemos que una relación se establece entre dos o más patrones. Cada patrón es un conjunto de objetos, referencias y valores. La estructura de un patrón puede expresarse mediante diagramas de objetos UML, aunque la especificación sugiere su extensión, con los elementos añadidos en los diagramas de transformación, como: la definición de dominios (clases, objetos y valores), las relaciones entre dominios (anotadas como C en el caso de **checkonly** o E en para **enforce**), elementos que aparecen en el primer recuadro de la transformación, así como las relaciones (*non-top*) a ejecutar en las cláusulas **when**, reflejadas en el segundo recuadro, y **where**, que son visibles en el tercer recuadro.

En definitiva, MDE va a ser un elemento fundamental en la propuesta de solución, al aportarnos las capacidades de modelado basado en metamodelos, así como los mecanismos de transformación entre modelos.

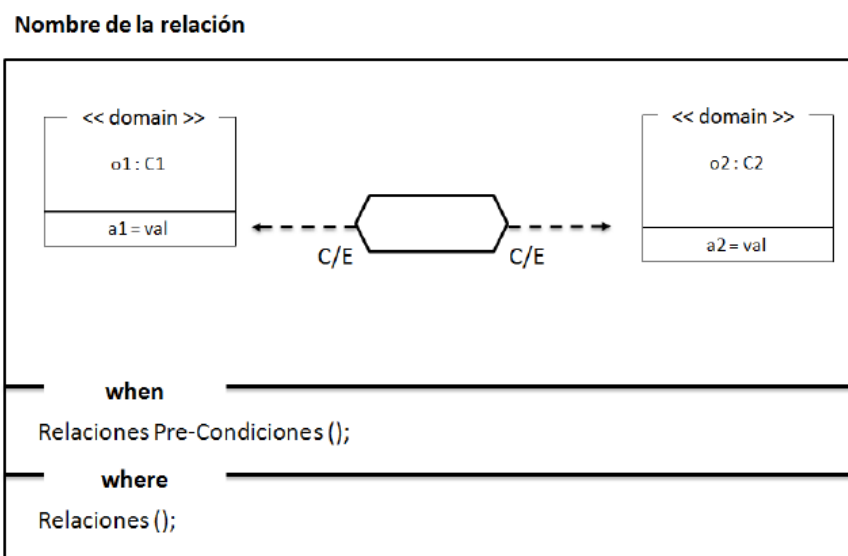


Figura 3.4: Sintaxis concreta gráfica de QVT Relations

3.4.3. La gestión de casos

Este trabajo está orientado a la colaboración entre procesos de diversas organizaciones, lo que en ocasiones se denominan procesos colaborativos, o inter organizacionales o procesos de la CdS. En este tipo de procesos, además de las particularidades propias de la colaboración definidas en la introducción (no existe actor protagonista en el proceso, es posible que una tarea deba realizarse entre varias organizaciones, se debe mantener la autonomía y privacidad entre las partes, etc), son procesos dinámicos que es muy posible que cambien.

Adicionalmente, es muy costoso incluir en el modelo de proceso todas las posibles modificaciones entre el modelo diseñado y el realmente ejecutado, ya que en ocasiones este tipo de colaboraciones vienen dirigidas por los datos o las interacciones, más que por el flujo de proceso. Llevado al límite es lo que se conoce en la literatura como procesos intensivos en conocimiento (KIPs), en los que la estructura no está completamente definida a priori y cobra especial importancia el papel de las personas que desarrollan el proceso y que toman decisiones atendiendo a la información del entorno y su experiencia[Hauder *et al.*, 2014]. Este tipo de procesos es fundamental para las organizaciones, atendiendo a Davenport [Davenport, 2013b], quien manifiesta que los procesos más importantes para las organizaciones hoy en día requieren conocimiento.

Los métodos tradicionales de gestión de flujos de trabajo no son soluciones válidas para dar soporte a este tipo de procesos [Van der Aalst *et al.*, 2003]. El enfoque tradicional para BPM se centra principalmente en los aspectos del proceso, dejando en un segundo lugar los datos o la información que se genera durante su ejecución [Hull *et al.*, 2013] Por ejemplo, BPMN asume que los procesos se ejecutan en base a una secuencia de actividades predeterminada y no da un soporte adecuado para controlar procesos en los que las decisiones las toman personas [de Man, 2009].

La gestión de casos (conocida como *case management* o *case handling*) surgió en la literatura de BPM en 2005 [Van der Aalst *et al.*, 2005] como un nuevo paradigma para dar soporte a los procesos de negocio intensivos en conocimiento y flexibles. Se trata de una forma de dar soporte al conocimiento de las personas en aquellos escenarios que requieren un nivel de flexibilidad mayor que los flujos de trabajo convencionales de los procesos de negocio [Marin *et al.*, 2013].

Las características de la gestión de casos son [de Man, 2009]:

- Los datos son el contexto central y compartido que dirige las actividades.
- Su ejecución está basada en eventos, en control de su estado interno y en una fuerte colaboración entre los participantes.
- El factor humano es predominante, ya que toma decisiones conforme se avanza en el caso, atendiendo a las alternativas previamente definidas .

Existen diversos paradigmas para modelar casos, distinguiendo entre los basados en actividades, artefactos y comunicaciones, si bien en la actualidad ninguno de ellos dispone de la capacidad de modelar el factor humano, esto es, el grado de libertad de las personas que intervienen en un caso para decidir qué actividades incluir en el mismo [de Man, 2009].

La gestión de casos adaptativos (*Adaptive Case Management, ACM*) hace referencia al trabajo necesario para dar soporte a un caso de una forma flexible, tomando decisiones en función del entorno y del contexto para seleccionar las alternativas de alguna forma se han predeterminado [Motahari-Nezhad y Swenson, 2013]. La revisión de la literatura llevada a cabo por Hauder *et al.* [Hauder *et al.*, 2014] concluye que la gestión de casos emerge como un paradigma válido para la gestión de KIPs, siendo ACM el enfoque que más atención está ganando en los campos de la investigación, la estandarización y la industria. Los autores destacan que siendo un campo inmaduro todavía no existe un modelo conceptual o una base teórica unificada para ACM.

El contexto del problema que pretendemos resolver, la creación de procesos colaborativos a partir de los modelos de procesos individuales de cada organización, coincide en algunas de las necesidades que se abordan desde la gestión de los casos, en particular lo relativo a poder representar en los modelos de procesos aspectos de planificación dinámica de los flujos, de manera que el proceso en colaboración pueda estar dirigido por los datos y las interacciones y no tanto por las secuencias de flujo. Por todo ello, dentro del marco CBG incluiremos un paso adicional que permita, si el escenario de modelado de CdS lo requiere, añadir la planificación dinámica pasando del dominio de procesos al de casos.

3.4.4. INROMA: un marco de referencia para facilitar la interoperabilidad y mantenibilidad de los modelos de procesos

INROMA (INteROperabilidad y MAntenibilidad) [García-Borgoñon, 2016] es una propuesta de lenguaje de modelado de procesos de software, concebido como un lenguaje auxiliar con el objetivo de facilitar la interoperabilidad y mantenibilidad de procesos de software. Orientado en su origen al ámbito de los procesos de software, sus elementos son de potencial aplicación a cualquier dominio de modelado de procesos y, al seguir el paradigma MDE y estar basado en UML, nos ofrece la característica de extensibilidad por lo que podemos ampliar sus elementos para adecuarlo a las necesidades de un dominio como el de la CdS.

En el ámbito del modelado de procesos software, tenemos una situación similar a la que nos hemos encontrado en el dominio de la CdS. Existen multitud de lenguajes de modelado de procesos software [García-Borgoñon *et al.*, 2014] sin que existe un lenguaje *ideal* o que se haya convertido en un estándar adoptado por la comunidad científica o la industria. Cada uno presenta unas propiedades y características mejores o peores para determinadas necesidades, sin que haya uno que sea mejor en cualquier escenario. Por ello, las organizaciones tienen serias dificultades a la hora de elegir qué lenguaje de modelado utilizar, quedando acoplados a esa elección por el que intentar utilizar otro lenguaje en otra circunstancia no es una tarea sencilla.

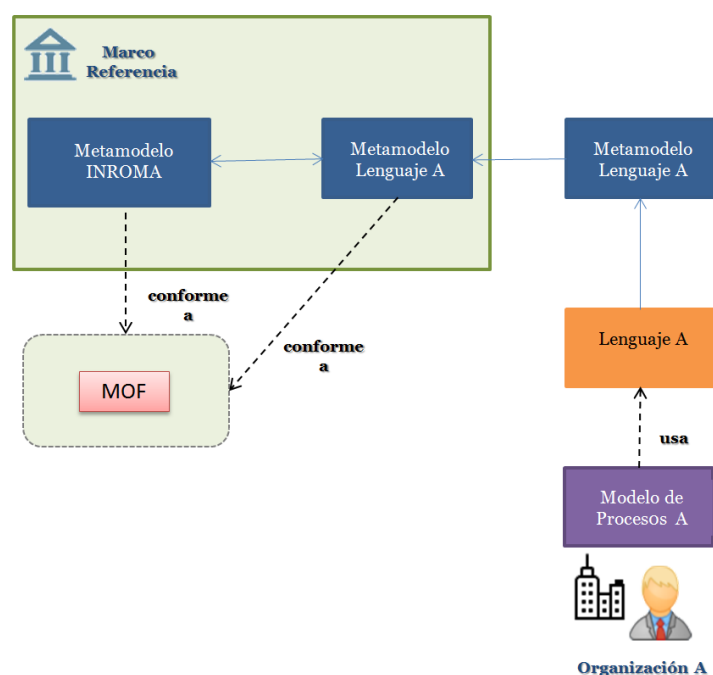


Figura 3.5: Detalle de metamodelos y transformaciones de INROMA

No existe un mecanismo que permita poder trabajar al mismo tiempo con procesos de software modelados con diferentes lenguajes, cada uno de ellos elegidos para satisfacer un requerimiento concreto para el que se considera más adecuado. La propuesta de García-Borgoñón [García-Borgoñón, 2016] se enfoca a romper este vínculo de forma que se facilite el uso y la interacción entre procesos de software modelados con lenguajes diferentes sin generar por ello inconsistencias o pérdidas de información. Para ello la autora desarrolla *un marco de referencia para facilitar, mejorar y agilizar la interoperabilidad y mantenibilidad de los modelos de procesos de software*, independientemente del lenguaje elegido para su modelado, basado en tres pilares, cuyo resumen a nivel de metamodelos y transformaciones queda reflejado en la figura 3.5:

- Un lenguaje de modelado de procesos de software denominado *INROMA*, sencillo y de fácil aprendizaje, concebido como lenguaje base que incorpora únicamente aquellos conceptos necesarios comunes para la definición y modelado de cualquier proceso de software.
- Un método mediante el que se establecen las bases teóricas que permiten incorporar nuevos lenguajes de modelado de procesos de software al marco de referencia.
- Un conjunto de transformaciones que se constituyen como el nexo entre cualquier lenguaje de modelado de procesos de software incorporado al marco de referencia e *INROMA*.

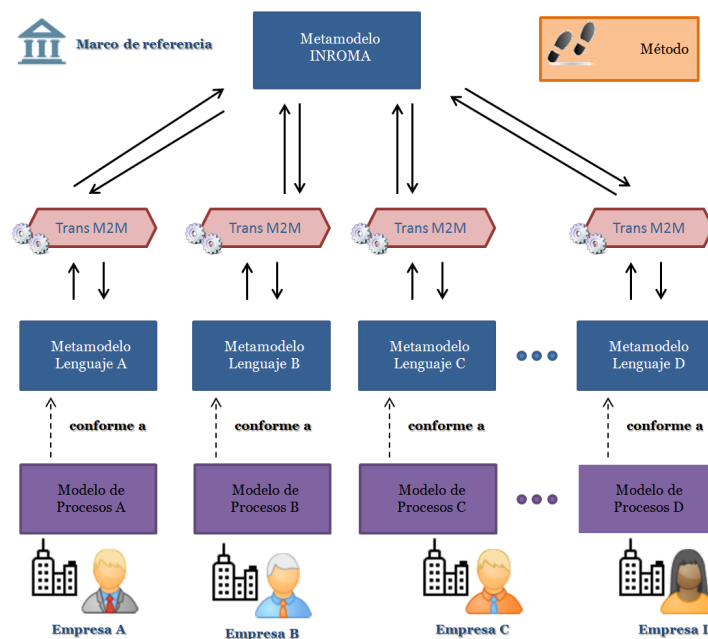


Figura 3.6: Vista del marco de referencia propuesto en INROMA

La figura 3.6 muestra el esquema de funcionamiento de la solución, en la que diversas organizaciones pueden crear modelos de sus procesos atendiendo al lenguaje de modelado que consideren más oportuno. Utilizando el método, se añade dicho lenguaje al marco de referencia, y se generan las transformaciones que permiten su conversión a un modelo de proceso conforme al metamodelo *INROMA*. De esta forma, con su adopción podemos unificar la descripción de los modelos con independencia del lenguaje utilizado en cada organización.

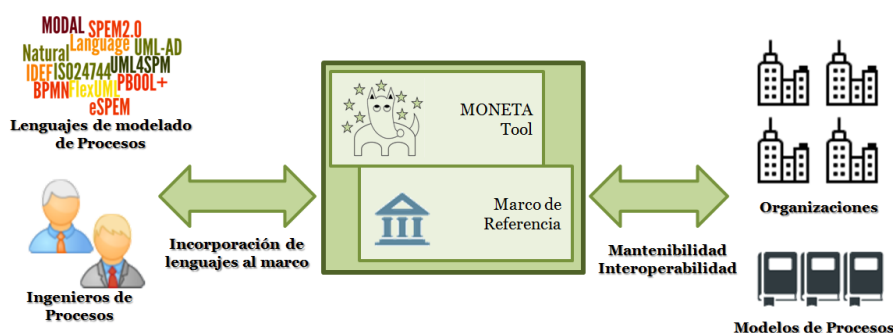


Figura 3.7: Vista del contexto del marco de referencia INROMA

La figura 3.7 muestra el contexto del marco de referencia. En ella se puede observar como, por un lado, es necesario incorporar los lenguajes de modelado de procesos de software que se deseen utilizar al marco de referencia. Por otro lado, los ingenieros de procesos pueden modelar sus procesos de software en el lenguaje de modelado de procesos de software que hayan elegido y que previamente haya sido incorporado al marco. Ambas acciones van a permitir y facilitar la interoperabilidad y mantenibilidad de los modelos de procesos definidos entre diferentes organizaciones. El marco contiene igualmente MONETA como una herramienta de soporte para su uso en la práctica.

La figura 3.8 muestra el metamodelo de INROMA, que tomaremos como base a la hora de llevar a cabo el enfoque bottom-up que nos permita reutilizar modelos de procesos de negocio creados con diversos lenguajes de modelado, para generar un modelo de interacción de procesos con mayor nivel de abstracción. Si bien deberemos extender la propuesta para incluir los elementos necesarios para modelar las interacciones entre los procesos, el marco de referencia creado va a constituir sin duda un buen elemento de partida para llevar a cabo nuestra propuesta.

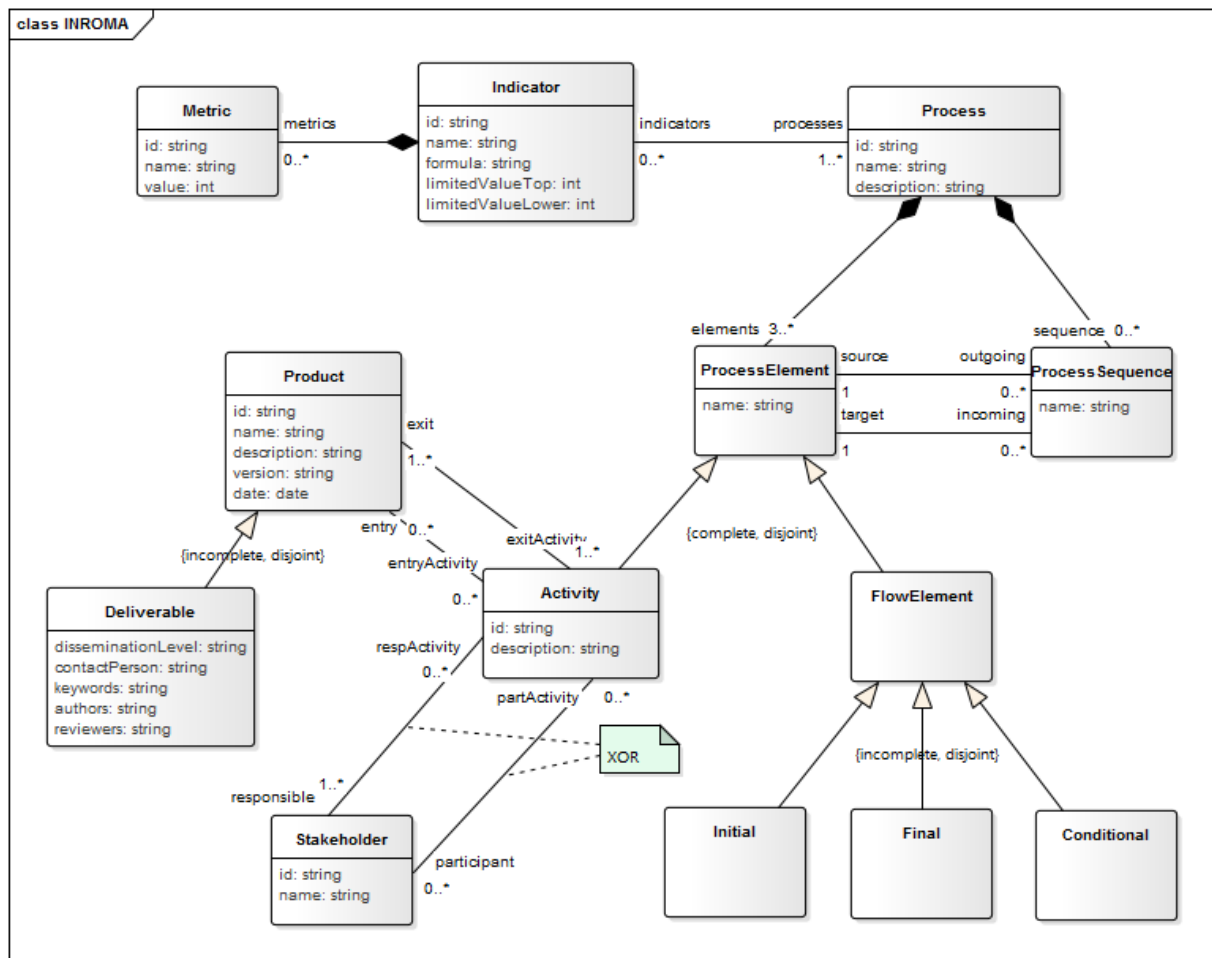


Figura 3.8: Metamodelo de interoperabilidad de lenguajes de procesos software

3.5. Estructura de la solución

Una vez hemos presentado los aspectos relevantes que determinan el problema a resolver y las principales influencias conceptuales y tecnológicas, en este apartado vamos a profundizar en la estructura que proponemos para abordar el problema de modelar los procesos de negocio de la CdS a partir de los modelos de procesos de negocio individuales que una organización pueda tener, con cualquier lenguaje de modelado de procesos, de forma que añadamos las capacidades de flexibilidad y adaptación de los modelos de procesos necesarias en el dominio de la CdS.

Nuestra propuesta de solución se basa en un marco de referencia denominado *Collaborative Business Generation (CBG)*, que pretende solventar el modelado de procesos y casos colaborativos con un enfoque bottom-up. Para ello, haciendo uso de nuevos metamodelos, métodos y transformaciones, seremos capaces de representar un modelo de proceso de una organización en: 1) una vista colaborativa del proceso, manteniendo la privacidad y la autonomía en la toma de decisiones y; 2) una vista del caso en colaboración, separando la parte estática del posible comportamiento dinámico.

El marco de referencia CBG está formado por:

- Un metamodelo, denominado *CBGProcess*, para dar soporte al modelado de procesos de negocio colaborativos.
- Una transformación M2M para generar la vista colaborativa desde un modelo de proceso individual.
- Un método para, a partir de la vista individual de un proceso conforme a *CBGProcess*, definir el proceso colaborativo. Dicho método permite separar la vista individual de un proceso de una organización de la vista colaborativa, manteniendo por tanto la privacidad y la autonomía en la toma de decisiones y compartiendo aquellos elementos que cada organización quiera para poder conformar el proceso en colaboración.
- Un metamodelo, denominado *CBGCase*, que siguiendo el paradigma de gestión de casos soporta el modelado de casos en entornos colaborativos.
- Una transformación M2M que permite, a partir de un proceso colaborativo conforme a *CBGProcess*, generar automáticamente una base de modelo de casos colaborativos conforme a *CBGCase*. De esta forma se consigue añadir la componente dinámica a los procesos, realizando una conversión del dominio de procesos al dominio de casos.
- Un método para, a partir del modelo de casos colaborativos generado, especificar qué elementos forman parte de la vista estática y cuáles en la vista dinámica del caso. De esta forma dispondremos del modelo de caso colaborativo generado a partir del proceso colaborativo.
- Una herramienta de soporte para facilitar la creación de modelos y la ejecución de las transformaciones desde un punto de vista práctico.

La figura 3.9 muestra un resumen de los pasos propuestos, en los que el primer paso es equivalente a hacer uso de la propuesta INROMA presentada en la sección anterior, el segundo paso permite la conversión a procesos colaborativos y el último paso lo transforma al dominio de los casos.

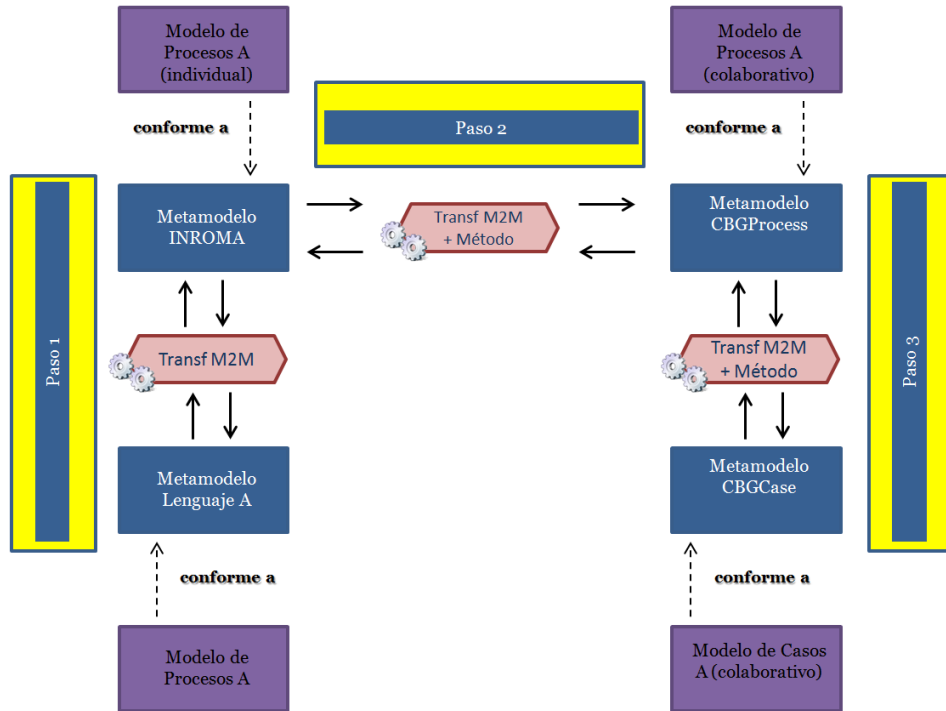


Figura 3.9: Resumen de la estructura de solución planteada en el marco CBG

Una vez hemos definido los elementos que van a formar parte de nuestra propuesta de solución, vamos a profundizar en cómo haremos uso de los mismos para abordar el problema planteado, en base a los siguientes pasos:

Paso 1

Siguiendo un enfoque bottom-up, a partir del modelo creado en cada organización con cada lenguaje, estableceremos una descripción del mismo con el lenguaje de modelado de procesos *INROMA* para unificar la visión de cada proceso con un lenguaje de modelado de procesos homogéneo. Como resultado dispondremos de un modelo de proceso de cada organización, atendiendo a su descripción individual.

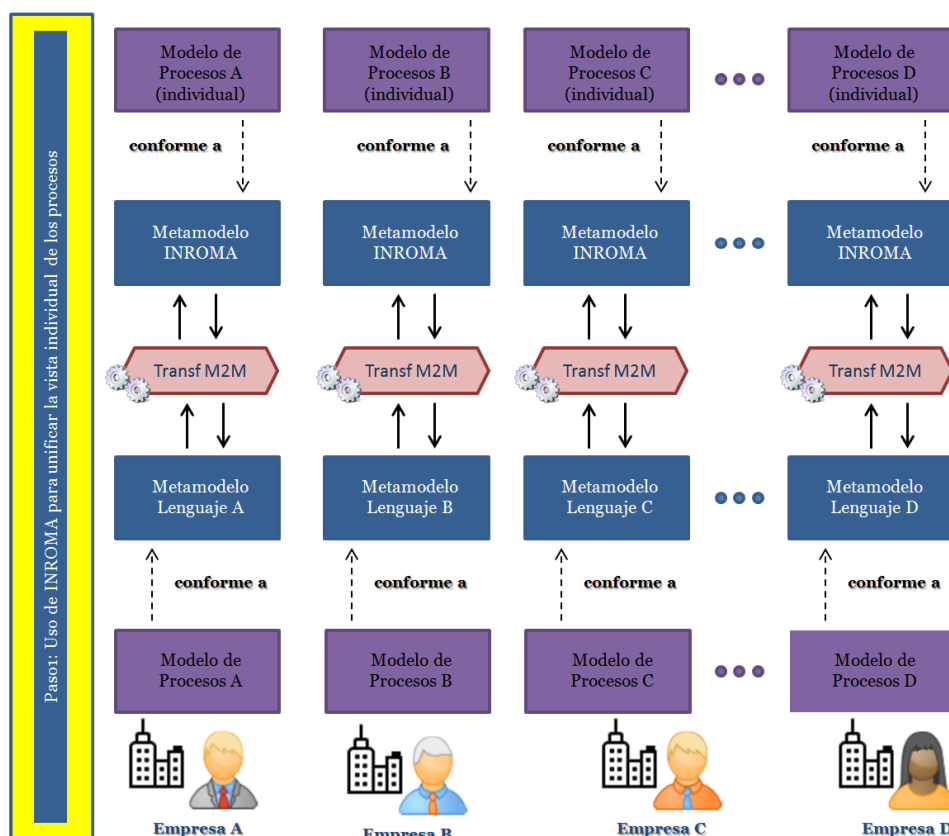


Figura 3.10: Vista del primer paso del marco CGB

Paso 2

A partir del modelo individual, siguiendo el método para definir una colaboración entre diversos procesos y utilizando el metamodelo *CBGProcess*, se genera un modelo de colaboración de procesos de negocio. Como resultado hemos separado la vista individual de un proceso de una organización de la vista colaborativa, manteniendo por tanto la privacidad y la autonomía en la toma de decisiones y compartiendo aquellos elementos que cada organización quiera para poder conformar el proceso en colaboración.

Este paso puede darse de dos formas. Por un lado, se puede crear un modelo para cada proceso de forma independiente, por lo que en nuestro caso se generarían cuatro modelos diferentes y cada uno se centraría en la relación con el resto, lo que sería un enfoque siguiendo el paradigma de la coreografía de servicios. Por otro lado, se puede seguir un enfoque de orquestación y componer

un único modelo en el que se detallen de forma global todas las relaciones.

Como resultado de este segundo paso, habremos separado la vista individual de un proceso de una organización de la vista colaborativa, manteniendo por tanto la privacidad y la autonomía en la toma de decisiones y compartiendo aquellos elementos que cada organización quiera para poder conformar el proceso en colaboración.

A través del método CGB, se generará el proceso colaborativo a partir de las vistas en colaboración de los procesos de cada organización, consiguiendo esa reutilización de modelos deseada con el enfoque bottom-up.

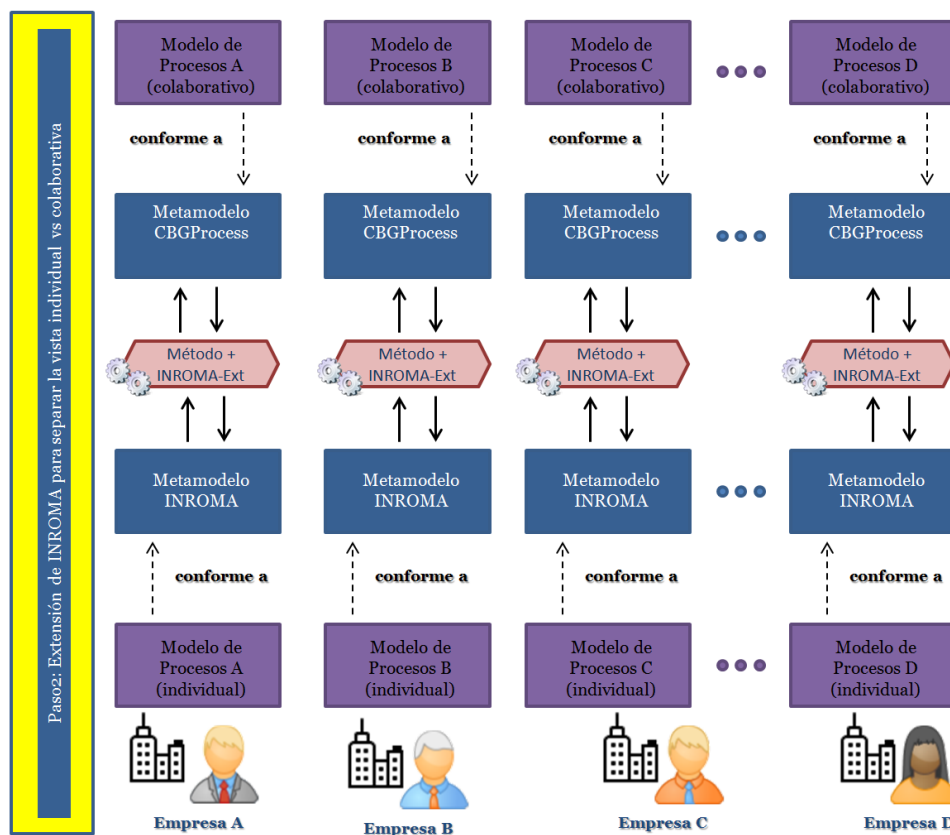


Figura 3.11: Vista del segundo paso del marco CGB

Paso 3

El último paso se dará únicamente en aquellos escenarios en los que la CdS requiera de planificación dinámica dentro de los procesos, realizando una conversión del dominio de procesos al dominio de casos. Para ello, a partir de una transformación M2M se genera automáticamente un modelo de casos colaborativos.

Tomando como fuente este modelo de casos en colaboración, que ha sido generado a partir del modelo de procesos colaborativo, se debe utilizar el método planteado para especificar qué elementos forman parte de la vista estática y cuáles en la vista dinámica del caso. Como resultado dispondremos del modelo de caso colaborativo conforme a *CBGCase*.

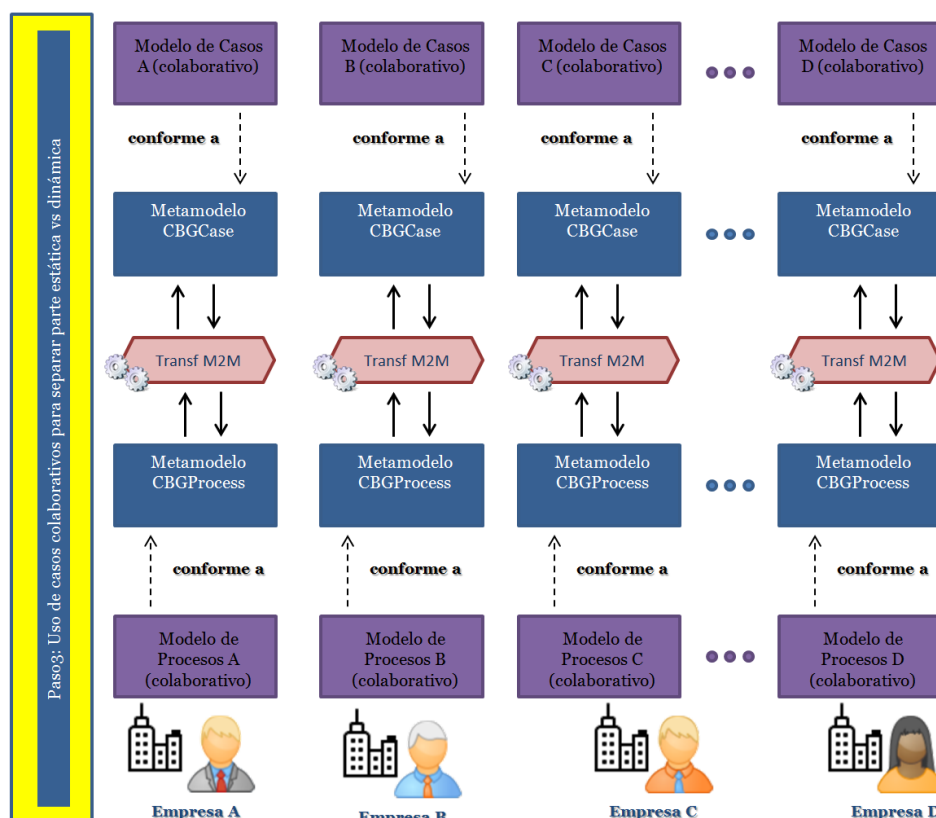


Figura 3.12: Vista del tercer paso del marco CGB

Como si de un proceso se tratara, enumeraremos los pasos a dar a nivel teórico, ya que para facilitar su adopción parte de estas acciones estarán soportadas de forma automática por una herramienta tal y como describiremos en el capítulo 7. La figura 3.13 presenta una vista del método.

En primer lugar hablaremos de las precondiciones, esto es, las condiciones que han de satisfacerse justo antes del comienzo del método, y que son las siguientes:

- Precondición 1 (PRE1): tenemos un escenario de proceso colaborativo a partir de un conjunto de procesos individuales de diversas organizaciones.

A partir de las precondiciones, el método queda definido por una secuencia de acciones, que gráficamente mostramos en el modelo de proceso 3.13 y que enumeramos a continuación:

- Actividad 1 (A1): en primer lugar se parte de los modelos de procesos individuales definidos en cualquier lenguaje y atendiendo a la solución planteada por García-Borgoñón [García-Borgoñón, 2016] se obtiene una descripción unificada de los procesos. Seguidamente, las actividades *A2* y *A3* resumen el método para generar el proceso colaborativo, que será detallado en la sección 4.3.
- Actividad 2 (A2): A partir de la descripción de los procesos individuales, se genera qué parte del proceso formará parte de la colaboración, es decir, qué elementos quiere cada participante hacer públicos y cuáles mantener bajo privacidad y autonomía. Como resultado tendremos un modelo, descrito conforme a *CBGProcess*, en el que se establecerá la vista pública de la colaboración.
- Actividad 3 (A3): A partir de esa vista pública de todos los procesos individuales se generará el *Modelo de proceso colaborativo*, primer *Deliverable* del método y equivalente a la *POST1*. Si en el dominio de aplicación no es requerida la parte dinámica, en cuanto a la planificación de actividades, el proceso finaliza. En otro caso, se realiza una conversión del dominio de procesos al de casos, de forma que las actividades *A4* y *A5* resumen el método para generar el caso colaborativo, que será detallado en la sección 5.3.
- Actividad 4 (A4): A partir del proceso colaborativo se realiza una transformación entre modelos que genera un primer caso colaborativo conforme a *CBGCase*, que incluye la posibilidad de establecer qué elementos del proceso son susceptibles de ser planificados en tiempo de ejecución.
- Actividad 5 (A5): Por último se finaliza la descripción del *Modelo de caso colaborativo*, segundo *Deliverable* y equivalente a la *POST2*.

Por último, como resultado del método, tendremos:

- Postcondición 1 (POST1): modelo de proceso colaborativo, conforme a *CBGProcess*, definido a partir de los procesos individuales de las organizaciones.
- Postcondición 2 (POST2): modelo de caso colaborativo, conforme a *CBGCase*, definido a partir del proceso colaborativo anteriormente definido (resultado opcional si el dominio de aplicación requiere de planificación dinámica).

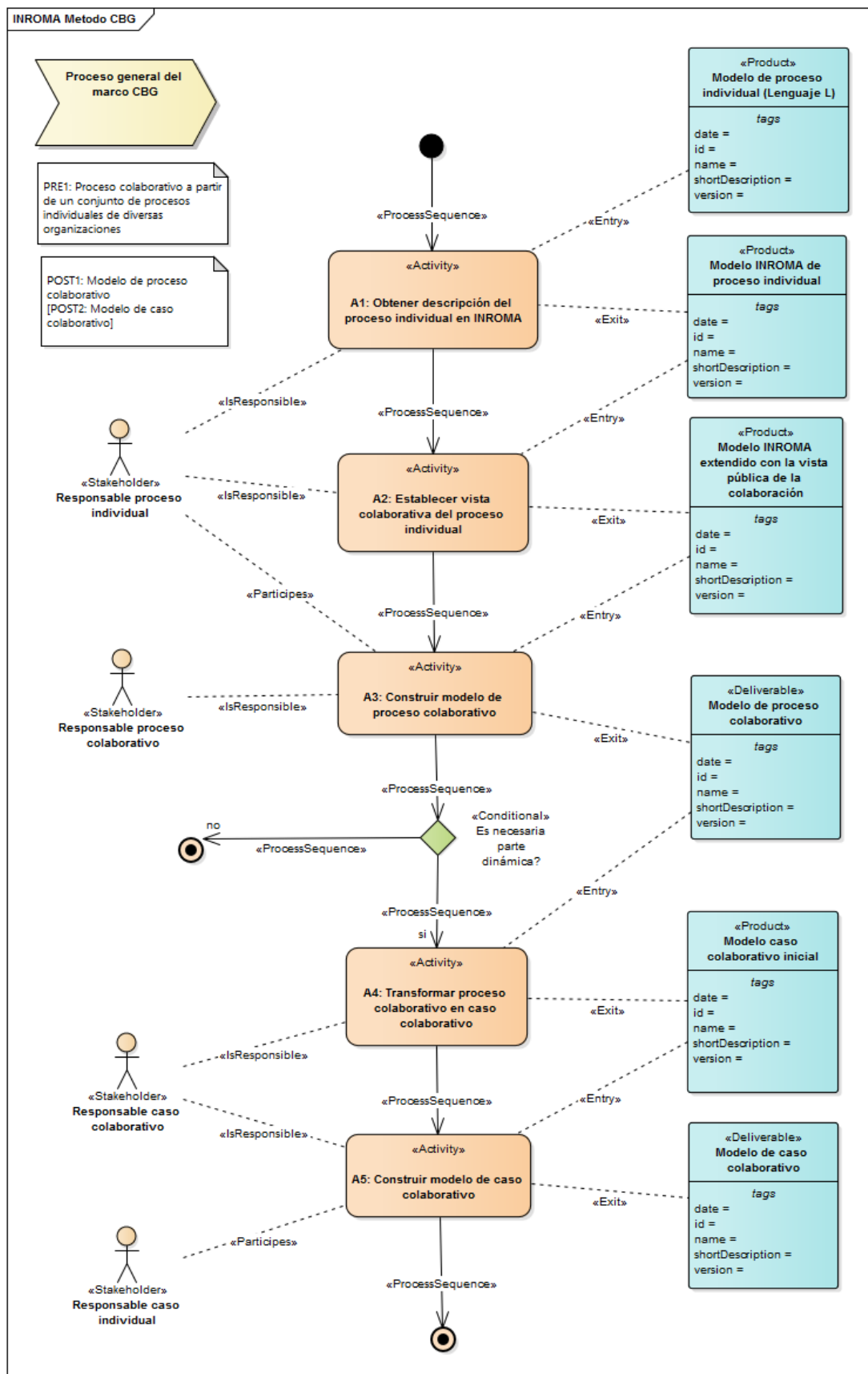


Figura 3.13: Proceso general del marco CGB

Por último, la tabla 3.1 resume cómo los diversos objetivos planteados en esta tesis son cubiertos con los elementos que forman parte del marco CGB.

Objetivo	Propuesta de solución
Realizar un estudio del estado del arte sobre lenguajes de modelado de procesos de la CdS	Revisión sistemática de la literatura
Definir un lenguaje de modelado de procesos de negocio colaborativos	1) Metamodelo <i>CBGProcess</i> para dar soporte al modelado de procesos de negocio colaborativos. 2) Transformación modelo a modelo (M2M) horizontal entre <i>INROMA</i> y <i>CBGProcess</i> para generar la vista colaborativa desde un proceso individual
Definir un método para, a partir de modelos de procesos individuales descritos en diversos lenguajes, poder construir el modelo de proceso colaborativo que representa la vista inter organizacional y las interacciones entre los mismos	Método para, a partir de un modelo conforme a <i>CBGProcess</i> , establecer qué elementos del mismo deben mantenerse privados (vista individual) y cuáles forman parte de la colaboración (vista colaborativa)
Definir un lenguaje de modelado de casos colaborativos	Metamodelo <i>CBGCase</i> que, siguiendo el paradigma de gestión de casos, soporta el modelado de casos en entornos colaborativos
Definir un marco conceptual para poder establecer el paso del dominio de procesos al de casos	1) Transformación M2M que permite, a partir de un proceso colaborativo conforme a <i>CBGProcess</i> , generar automáticamente una base de modelo de casos colaborativos conforme a <i>CBGCase</i> . 2) Método para, a partir del modelo de casos colaborativos generado, concretar qué elementos forman parte de la vista estática y cuáles en la vista dinámica del caso
Desarrollar una herramienta que dé soporte para que este marco conceptual que pueda ser utilizado en la práctica por organizaciones	Una herramienta de soporte, denominada <i>CBG-Tool</i> , para facilitar la creación de modelos y la ejecución de las transformaciones desde un punto de vista práctico
Validar la propuesta con casos de estudio reales	Aplicación del marco <i>CBG</i> en proyectos reales

Tabla 3.1: Relación entre los objetivos planteados y la estructura de la solución propuesta

3.6. Conclusiones

En este capítulo se ha presentado con mayor detalle el planteamiento del problema y la propuesta de solución. Para ello, hemos puesto en contexto el dominio del problema, afirmando que las organizaciones necesitan definir y mejorar los procesos de su cadena de valor, analizando que la mejora de los procesos de la CdS requiere de entornos basados en modelos flexibles y adaptables pero que para crear esos modelos existen muchos lenguajes de modelado. Seguidamente hemos definido la necesidad de fondo que perseguimos, poder disponer de una solución que permita crear modelos de procesos de CdS, reutilizando los modelos de los procesos que cada organización pueda haber creado con diversos lenguajes, y que hemos hecho explícita mediante tres casos de uso. Tras enumerar los objetivos que nos planteamos, hemos repasado las principales influencias conceptuales y tecnológicas que nos han marcado, tanto las lecciones aprendidas de la revisión sistemática de la literatura, como los paradigmas basados en modelos y servicios o la propuesta para facilitar la interoperabilidad de los modelos de procesos conformes a cualquier lenguaje. Por último hemos detallado la propuesta de solución que planteamos, el marco de referencia *Collaborative Business Generation (CBG)*, para dar soporte al modelado de procesos y casos colaborativos con un enfoque bottom-up, y que está formado por:

- Un metamodelo, denominado *CBGProcess*, para dar soporte al modelado de procesos de negocio colaborativos.
- Una transformación M2M para generar la vista colaborativa desde un modelo de proceso individual.
- Un método para, a partir de la vista individual de un proceso conforme a *CBGProcess*, definir el proceso colaborativo. Dicho método permite separar la vista individual de un proceso de una organización de la vista colaborativa, manteniendo por tanto la privacidad y la autonomía en la toma de decisiones y compartiendo aquellos elementos que cada organización quiera para poder conformar el proceso en colaboración.
- Un metamodelo, denominado *CBGCase*, que siguiendo el paradigma de gestión de casos soporta el modelado de casos en entornos colaborativos.
- Una transformación M2M que permite, a partir de un proceso colaborativo conforme a *CBGProcess*, generar automáticamente una base de modelo de casos colaborativos conforme a *CBGCase*. De esta forma se consigue añadir la componente dinámica a los procesos, realizando una conversión del dominio de procesos al dominio de casos.
- Un método para, a partir del modelo de casos colaborativos generado, especificar qué elementos forman parte de la vista estática y cuáles en la vista dinámica del caso. De esta forma dispondremos del modelo de caso colaborativo generado a partir del proceso colaborativo.
- Una herramienta de soporte para facilitar la creación de modelos y la ejecución de las transformaciones desde un punto de vista práctico.

Seguidamente pasamos a exponer cada uno de los elementos que forman parte del marco *CBG* con mayor detalle.

Capítulo 4

Lenguaje y método para el modelado de procesos colaborativos a partir de modelos de procesos individuales

En el capítulo anterior se ha definido con detalle el planteamiento del problema y la propuesta de solución, que cubre la necesidad que tienen las organizaciones de definir y mejorar los procesos de su cadena de valor, destacando la flexibilidad y adaptabilidad como características propias del dominio de la CdS. Este problema ha quedado manifestado en la necesidad de disponer de una solución que permita crear modelos de procesos colaborativos en una CdS, reutilizando los modelos de los procesos que cada organización pueda haber creado a nivel individual con diversos lenguajes. Dado que en el ámbito de la CdS es posible que la decisión de qué actividades llevar a cabo sea tomada por una persona de forma dinámica, nuestro enfoque llega al dominio de casos, añadiendo la planificación dinámica a la vista estática, y lo extiende con el carácter colaborativo propio de la CdS.

La propuesta de solución, que hemos denominado marco de referencia *Collaborative Business Generation (CBG)*, se enfoca al modelado de procesos y casos colaborativos con un enfoque bottom-up, y se compone de metamodelos, transformaciones y métodos para generar esos modelos. En este capítulo detallaremos sus primeros componentes:

- Un metamodelo, denominado *CBGProcess*, para dar soporte al modelado de procesos de negocio colaborativos.
- Un método para, a partir de la vista individual de un proceso conforme a *CBGProcess*, definir el proceso colaborativo.

El capítulo comienza con la exposición de las necesidades no cubiertas por el metamodelo INROMA y que justifican su extensión. En la segunda sección se describe el nuevo metamodelo en su globalidad. En la tercera sección se detalla el método para, a partir de un modelo INROMA, realizar la adaptación sobre el metamodelo extendido para dar soporte a un modelo con su visión colaborativa. Por último, se presenta un resumen con las principales conclusiones del capítulo.

4.1. Necesidades que justifican la extensión de INROMA

Seguidamente pasamos a exponer qué elementos no pueden ser cubiertos con la propuesta actual en el contexto del dominio del problema que pretendemos resolver y que justifican las extensiones que serán detalladas en la siguiente sección. Como hemos detallado en la sección 3.2, la necesidad de fondo que perseguimos es poder disponer de una solución que permita crear modelos de procesos de CdS, reutilizando los modelos de los procesos que cada organización pueda haber creado con sus lenguajes. INROMA ofrece el primer paso en cuanto a la capacidad de, partiendo de modelos de procesos conformes a diversos lenguajes, generarlos en base a un lenguaje común de forma automatizada. Si bien de esta forma garantizamos que en los procesos que forman parte de una CdS pueden ser modelados con un lenguaje auxiliar común, como es INROMA, su versión actual no da soporte para expresar los puntos relativos a la colaboración entre los mismos, manteniendo la privacidad y autonomía individuales. Seguidamente detallamos aquellos conceptos que no pueden ser cubiertos y que justifican la necesidad de extender el metamodelo INROMA.

- Vista multi organizacional: INROMA está planteado para modelar procesos de software de una organización, pero en nuestro contexto es requisito imprescindible poder asociar los procesos a una organización, siendo este enfoque inter organizacional uno de los elementos clave del problema a resolver.
- Actividades en colaboración: con la versión actual, no es posible identificar que una tarea debe ser realizada por varios participantes, más allá de poder asociarla manteniendo su mismo nombre, pero perderíamos la semántica que implica anotar una tarea como colaborativa entre múltiples participantes y/o procesos. Por otro lado queremos mantener la trazabilidad entre la vista del modelo individual y la colaborativa, por lo que en el caso de querer hacer pública una actividad es preciso mantener las relaciones con otras actividades o productos que se mantengan de forma privada a la organización.
- Productos en colaboración: fruto de una actividad que es desarrollada por varios actores, su resultado es a su vez un producto que puede ser entrada y/o salida de actividades en otros procesos, constituyendo en sí mismo un producto colaborativo, siendo otra necesidad no cubierta en la actualidad por INROMA. Estos productos representan la *Process Collaboration Information*) en nuestro marco teórico.
- Indicadores y métricas en colaboración: uno de los principales objetivos de una colaboración entre procesos de diversos actores es sin duda obtener mejores resultados en su conjunto que si se desarrollan de forma individual, por lo que es preciso disponer de indicadores de colaboración, denominados *Collaborative Process Performance Indicator* en nuestro marco teórico.

En nuestro contexto se requiere un modelado de procesos colaborativos en un contexto multi sectorial, dado que como hemos presentado en la introducción, la CdS implica procesos de cualquier dominio. INROMA fue diseñado para el modelado de procesos de software, si bien en su validación [García-Borgoñon, 2016] quedó de manifiesto que el resultado alcanzado con INROMA como mecanismo de interoperabilidad no es sólo factible para el ámbito del software, sino que se hacía extensible a nuevos entornos de aplicación. En cualquier caso se han analizado las necesidades relativas al dominio de la CdS para terminar comprobando que todos los conceptos pueden quedar recogidos en su propuesta actual. A la vista de estas necesidades y, haciendo uso del mecanismo de extensibilidad de UML sobre el que INROMA está fundamentado y que se muestra en la figura 4.1, pasamos a detallar los nuevos elementos que forman parte de la extensión propuesta en este trabajo.

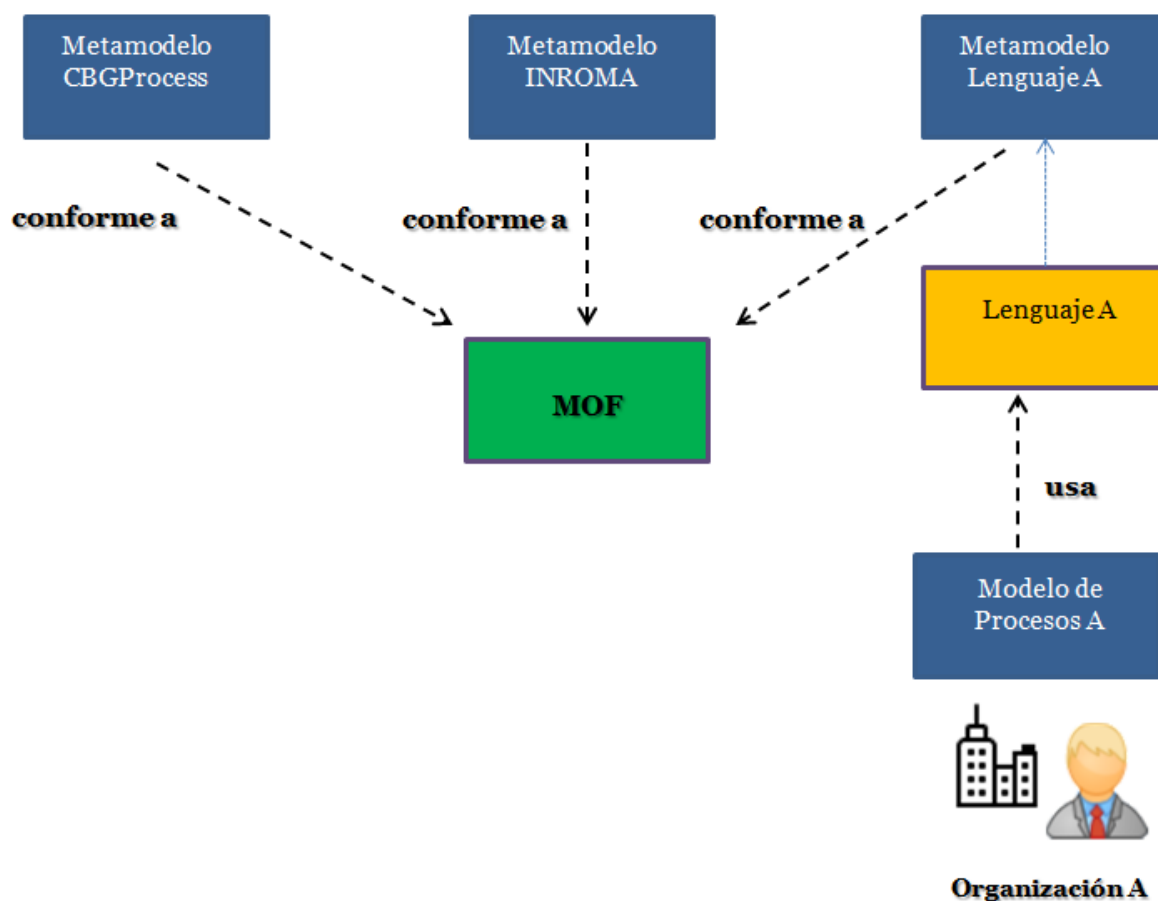


Figura 4.1: Uso del mecanismo de extensibilidad de UML sobre INROMA

4.2. Metamodelo *CBGProcess*

En esta sección se presenta la extensión del metamodelo INROMA para el modelado de procesos colaborativos, atendiendo a las necesidades no cubiertas expuestas en la sección anterior. La figura 4.2 muestra dicho metamodelo, en la que se han añadido los siguientes elementos:

- La metaclase *BlackBoxActivity* constituye aquellas actividades del proceso que desean mantenerse privadas a la organización y que no formarán parte como tal del proceso colaborativo.
- La metaclase *BlackBoxIndicator* representa la forma en la que una organización desea medir de forma individual un proceso a partir de las métricas asociadas.
- La metaclase *BlackBoxMetric* representa una medida cuantitativa que nos va a permitir evaluar un determinado aspecto del proceso a nivel individual.
- La metaclase *BlackBoxProduct* representa cualquier pieza de información consumida (entrada), producida (salida) o modificada (entrada y salida) durante la ejecución de una actividad que se va a mantener privada fuera del proceso colaborativo.
- La metaclase *Organization* representa una organización que forma parte de una CdS sobre la que queremos modelar los procesos en colaboración.

Adicionalmente, existe un conjunto de elementos de INROMA cuya semántica cambia, ya que pasan a representar elementos de un proceso en colaboración:

- La metaclase *Activity*: representa una actividad de un proceso colaborativo.
- La metaclase *Indicator* representa la forma en la que todas las organizaciones a nivel global desean medir un proceso a partir de las métricas colaborativas asociadas.
- La metaclase *Metric* representa una medida cuantitativa que nos va a permitir evaluar un determinado aspecto del proceso a nivel colaborativo.
- La metaclase *Product*: cualquier pieza de información consumida (entrada), producida (salida) o modificada (entrada y salida) durante la ejecución de una actividad colaborativa.

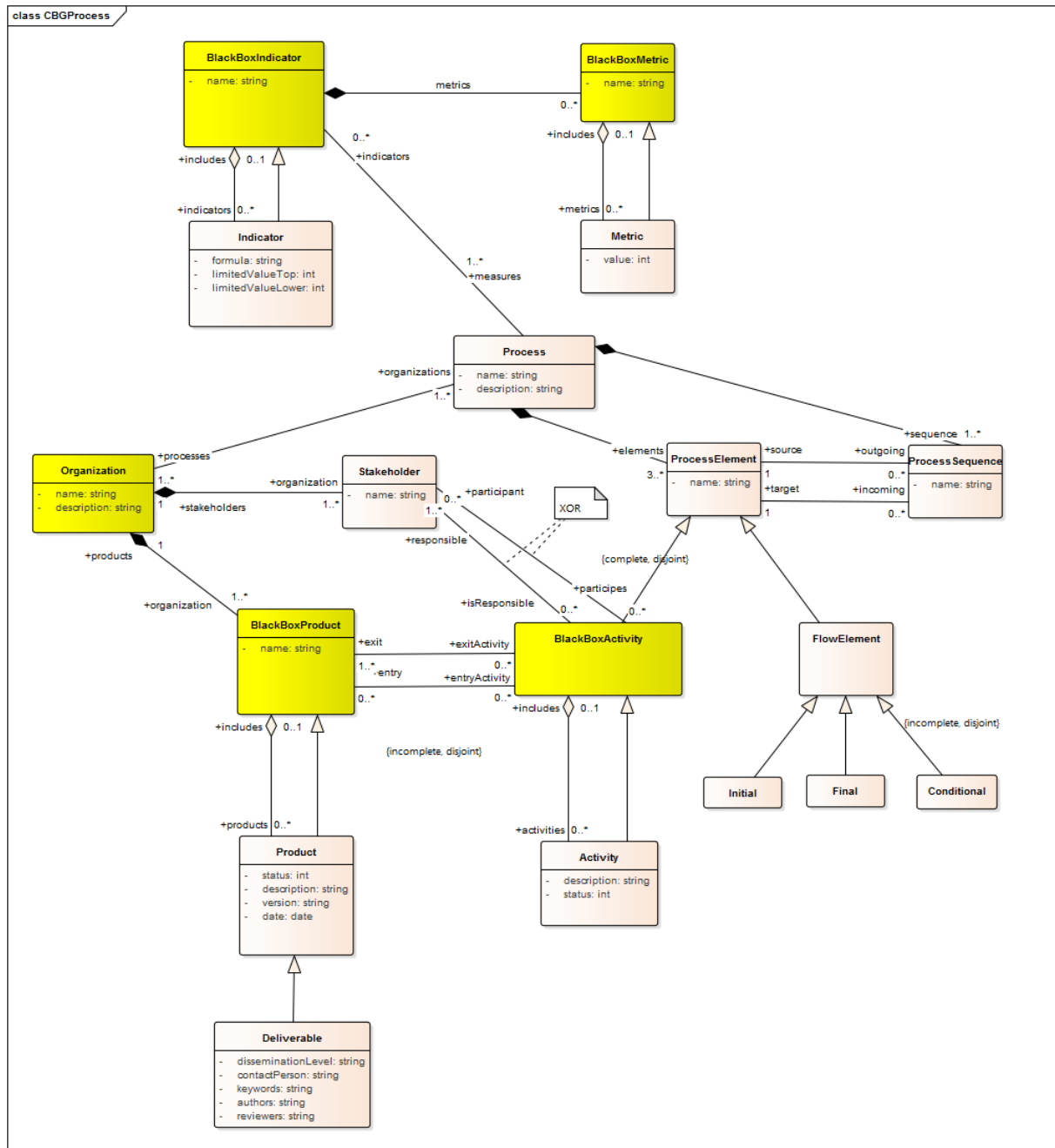


Figura 4.2: Metamodelo *CBGProcess* para el modelado de procesos colaborativos

Restricciones

Todo proceso debe respetar los siguientes requerimientos mínimos:

- Son procesos de entrada única - existe exactamente un único elemento de inicio en el proceso. El fragmento OCL 5.1 describe la restricción.

```

1 context Process
2 inv exactlyOneStart : (self.processElement → select(os|isTypeOf(Initial))) →
   size() = 1

```

Restricción 4.1: Restricción sobre la metaclase *Process*: El proceso tiene un elemento inicial

- Son procesos de múltiples salidas - al menos hay un elemento final en el proceso. El fragmento OCL 5.2 describe la restricción.

```

1 context Process
2 inv multipleEnds : (self.processElement → select(os|isTypeOf(Final))) → size
   () >= 1

```

Restricción 4.2: Restricción sobre la metaclase *Process*: El proceso tiene al menos un elemento final

- Son procesos que tienen al menos una actividad El fragmento OCL 5.3 describe la restricción.

```

1 context Process
2 inv atLeastOneActivity : (self.processElement → select(os|isTypeOf(Activity))
   ) → size() >= 1

```

Restricción 4.3: Restricción sobre la metaclase *Process*: El proceso tiene al menos una actividad

A continuación ofrecemos una explicación detallada de cada una de las metaclases que lo constituyen en orden alfabético. Para cada una de ellas se presenta su descripción, atributos, asociaciones, y sus generalizaciones directas y restricciones si existieran, atendiendo a la misma definición que el propio UML.

4.2.1. Activity

Descripción La metaclasses *Activity* describe cualquier fragmento de trabajo a ser ejecutado durante el proceso colaborativo. Dentro de una CdS, esta metaclasses hará referencia a los elementos públicos de la colaboración o, en la vista individual de un modelo de proceso colaborativo, a los elementos que se comparten entre varias organizaciones.

Generalización *BlackBoxActivity*

Atributos

- *name:String* (de *ProcessElement*) El nombre de la actividad
- *description:String* Ofrece una breve descripción de la actividad
- *status:int* Representa el estado de la actividad

Asociaciones

- *includes:BlackBoxActivityProduct[0..1]* En la vista individual una actividad pública podrá estar relacionada con una actividad individual

Restricciones Ninguna

4.2.2. BlackBoxActivity

Descripción La metaclasses *BlackBoxActivity* describe cualquier fragmento de trabajo a ser ejecutado durante un proceso individual o en la vista individual del proceso colaborativo. Dentro de una CdS, esta metaclasses hará referencia a los elementos privados de la colaboración o, en la vista individual de un modelo de proceso colaborativo, a los elementos que no se comparten entre varias organizaciones.

Generalización *ProcessElement*

Atributos

- *name:String* (de *ProcessElement*) El nombre de la actividad individual

Asociaciones

- *activities:Activity[0..*]* Conjunto de actividades colaborativas que forman parte de la vista individual de la actividad

- *entry:BackBoxProduct[0..*]* Conjunto de productos de entrada para la ejecución de esta actividad
- *exit:BackBoxProduct[1..*]* Conjunto de productos de salida que se obtienen como resultado de la ejecución de esta actividad
- *participant:Stakeholder[0..1]* Rol que participa en la ejecución de una actividad pero no es el responsable de la misma
- *responsible::Stakeholder[0..*]* Rol (o roles) responsable de la ejecución de una actividad

Restricciones Ninguna

4.2.3. BlackBoxIndicator

Descripción La metaclase *BlackBoxIndicator* representa un indicador de proceso, a nivel individual, que se establece a partir de una métrica y los valores límite objetivo para poder evaluar si el proceso ha cumplido o no dicho indicador.

Generalización Ninguna

Atributos

- *name:String* El nombre del indicador

Asociaciones

- *indicators:Indicator[0..*]* Conjunto de indicadores colaborativos que forman parte de la vista individual del indicador
- *measures:Process[1..*]* Conjunto de procesos sobre los que actúa un indicador
- *metrics:BlackBox[0..*]* Conjunto de métricas que forman parte del indicador

Restricciones

Ninguna

4.2.4. BlackBoxMetric

Descripción La metaclase *BlackBoxMetric* representa una medida cuantitativa que nos va a permitir evaluar un determinado aspecto del proceso a nivel individual

Generalización Ninguna

Atributos

- *name:String* El nombre de la métrica

Asociaciones

- *metrics:Metric[0..*]* Conjunto de métricas colaborativas que forman parte de la vista individual de la métrica

Restricciones Ninguna

4.2.5. BlackBoxProduct

Descripción La metaclass *BlackBoxProduct* representa cualquier pieza de información consumida (entrada), producida (salida) o modificada (entrada y salida) durante la ejecución de una actividad individual dentro del proceso colaborativo.

Generalización Ninguna

Atributos

- *name:String* El nombre del elemento del producto individual

Asociaciones

- *entry:BlackBoxActivity[0..*]* Un producto individual puede ser un elemento de entrada para una o varias actividades individuales
- *exit:BlackBoxActivity[0..*]* Un producto individual puede ser un elemento de salida para una o varias actividades individuales
- *products:Product[0..*]* Conjunto de productos colaborativos que forman parte de la vista individual del producto

Restricciones Ninguna

4.2.6. Conditional

Descripción La metaclass *Conditional* permite establecer ramas excluyentes en la secuencia lógica de actividades.

Generalización FlowElement

Atributos

- *name:String* (de *ProcessElement*) El nombre del elemento condicional

Asociaciones Ninguna

Restricciones La restricción que se establece sobre un elemento *Conditional* es que debe tener al menos dos enlaces de salida y, al menos, uno de entrada. El fragmento OCL 5.5 describe la restricción.

```

1 context ProcessElement
2 inv: self.isTypeOf (Conditional) implies (self.outgoing → size()>=2 and self.
    incoming → size()>=1)

```

Restricción 4.4: Restricción sobre la metaclass *Conditional*: al menos, dos enlaces de salida y, al menos, uno de entrada

4.2.7. Deliverable

Descripción La metaclass *Deliverable* representa un tipo concreto de producto que es entregable. Es importante distinguir entre los productos que son artefactos transitorios dentro de un proceso y los que son entregables, que representan el resultado tangible de una actividad. Un producto entregable requiere de mayor atención por parte de los roles participantes y, sobre todo, del responsable de la actividad.

Generalización Product

Atributos

- *authors:String* Los autores del entregable
- *date:String* (de *Product*) La fecha del deliverable que especifica la última vez que fue modificado
- *contactPerson:String* La persona de contacto en todo aquello que haga referencia al entregable

- *description:String* (*de Product*) Una descripción del deliverable
- *disseminationLevel:String* Establece si se trata de un entregable público o confidencial
- *keywords:String* Las palabras claves por las que el entregable es indexado para posibles búsquedas
- *name:String* (*de BackBoxProduct*) El nombre del deliverable
- *reviewers:String* Los revisores del entregable
- *status:int* (*de Product*) Representa el estado como un mecanismo de sincronización de información entre los colaboradores
- *version:String* (*de Product*) La versión del deliverable

Asociaciones

Ninguna

Restricciones

Ninguna

4.2.8. Final

Descripción La metaclase *Final* representa el punto de salida de un proceso, tras el cual podemos decir que se ha finalizado.

Generalización FlowElement

Atributos

- *name:String* (*de ProcessElement*) El nombre del elemento final

Asociaciones Ninguna

Restricciones El elemento final de un proceso no puede tener ningún enlace de salida. El fragmento OCL 5.6 describe la restricción.

```

1 context Process
2 inv noOutgoing : (self.processElement → select(os|isTypeOf(Final))).outgoing →
    size() = 0

```

Restricción 4.5: Restricción sobre la metaclase *Final*: El elemento final no tiene enlace de salida

4.2.9. FlowElement

Descripción La metaclasses *FlowElement* representa los elementos que forman parte de la estructura de un proceso pero no se corresponden a la ejecución de ningún fragmento de trabajo.

Generalización *ProcessElement*

Atributos

- *name:String* (de *ProcessElement*) El nombre del elemento de flujo

Asociaciones Ninguna

Restricciones Ninguna

4.2.10. Indicator

Descripción La metaclasses *Indicator* representa un indicador de proceso colaborativo que se establece a partir de una métrica y los valores límite objetivo para poder evaluar si el proceso ha cumplido o no dicho indicador. Es un elemento básico para cualquier cuadro de mando y es exigido por la mayor parte de los estándares de calidad de procesos.

Generalización *BlackBoxIndicator*

Atributos

- *name:String* (de *BlackBoxIndicator*) El nombre del indicador que constituye su identificador para el usuario
- *formula:String* Establece la formula con la que se va a obtener la métrica
- *limitedValueTop:String* Establece el límite superior establecido para la evaluación del indicador
- *limitedValueLower:String* Establece el límite inferior del valor para la evaluación del indicador

Asociaciones

- *includes:BlackBoxIndicator[0..1]* En la vista individual un indicador público puede estar relacionado con un indicador individual

Restricciones

Ninguna

4.2.11. Initial

Descripción La metaclase *Initial* representa el punto de entrada al proceso, a partir del cual se secuencian el resto de los elementos de proceso, pero no corresponde a la ejecución de ningún fragmento de trabajo.

Generalización FlowElement

Atributos

- *name:String* (de *ProcessElement*) El nombre del elemento de inicio

Asociaciones Ninguna

Restricciones El elemento inicial de un proceso no puede tener ningún enlace de entrada. El fragmento OCL 5.7 describe la restricción.

```

1 context Process
2 inv noIncoming : (self.processElement → select(os|isTypeOf(Initial))).incoming →
    size() = 0
  
```

Restricción 4.6: Restricción sobre la metaclase *Initial*: El elemento inicial no tiene enlace de entrada

4.2.12. Metric

Descripción La metaclase *Metric* representa una medida cuantitativa que nos va a permitir evaluar un determinado aspecto del proceso colaborativo.

Generalización BlackBoxMetric

Atributos

- *name:String* (de *BlackBoxMetric*) El nombre de la métrica, que constituye un identificador único de la misma
- *value:int* El valor de la métrica actual

Asociaciones

- *includes:BackBoxMetric[0..1]* En la vista individual una métrica pública puede estar relacionada con una métrica individual

Restricciones Ninguna

4.2.13. Organization

Descripción La metaclase *Organization* representa una organización que forma parte de una CdS sobre la que queremos modelar los procesos en colaboración. Cabe destacar que cuando estamos hablando de organizaciones distintas en una cadena de valor, bien podríamos estar en el caso de grandes organizaciones que tienen áreas funcionales en su estructura, como pueden ser departamentos de compras, producción y logística, encargados desde el aprovisionamiento hasta la entrega de los productos al cliente. Dado que en la literatura está ampliamente definida la CdS como la participación de dos o más entidades independientes, entendidas como organizaciones jurídicamente distintas, aunque en la definición del problema hemos mantenido esta suposición en el metamodelo daremos cabida tanto a organizaciones distintas como a diversas áreas funcionales bajo una misma empresa.

Generalización Ninguna

Atributos

- *name:String* El nombre de la organización
- *description:String* Ofrece una descripción de la organización y su rol en la cadena de valor

Asociaciones

- *products:BlackBoxProduct[0..*]* Conjunto de productos colaborativos de la organización
- *processes:Process[1..*]* Conjunto de procesos de la organización
- *stakeholders:Stakeholder[0..*]* Conjunto de stakeholders de la organización

Restricciones Ninguna

4.2.14. Process

Descripción La metaclase *Process* es la clase principal del metamodelo. De acuerdo a la definición de proceso introducida en el primer capítulo, representa un conjunto ordenado de

acciones que son ejecutadas por alguien con el objetivo de producir una serie de resultados o productos.

Generalización Ninguna

Atributos

- *name:String* El nombre del proceso.
- *description:String* Ofrece una descripción del proceso de forma textual, incluyendo aspectos tales como la finalidad del mismo o los objetivos que debe cumplir.

Asociaciones

- *organizations:Organization[1..*]* Establece las organizaciones que forman parte del proceso
- *elements:ProcessElement[3..*] ordered* Conjunto ordenado de los elementos que forman parte de la estructura de un proceso
- *sequence:ProcessSequence[1..*]* Conjunto de conexiones que muestran el flujo entre los elementos del proceso
- *indicators:Indicator[0..*]* Conjunto de indicadores que definen los aspectos a medir de un proceso

Restricciones Ninguna

4.2.15. ProcessElement

Descripción La metaclase *ProcessElement* representa los principales elementos que conforman la estructura de un proceso, es decir, los pasos a seguir para la ejecución del mismo.

Generalización Ninguna

Atributos

- *name:String* El nombre del elemento del proceso

Asociaciones

- *elements:Process[1]* Establece el proceso del que forma parte este elemento
- *incoming:ProcessSequence[0..*]* Conexión que tiene el elemento de proceso como destino

- *outgoing:ProcessSequence[0..*]* Conexión que tiene el elemento de proceso como origen

Restricciones Ninguna

4.2.16. ProcessSequence

Descripción La metaclassa *ProcessSequence* establece las conexiones entre dos elementos de proceso.

Generalización Ninguna

Atributos

- *name:String* El nombre de la secuencia del proceso

Asociaciones

- *sequence:Process[1]* Establece el proceso del que forma parte esta conexión de secuencia
- *source:ProcessElement[1]* Elemento de proceso desde el que parte la conexión
- *target:ProcessElement[1]* Elemento de proceso hacia el que se dirige la conexión

Restricciones Ninguna

4.2.17. Product

Descripción La metaclassa *Product* representa cualquier pieza de información consumida (entrada), producida (salida) o modificada (entrada y salida) durante la ejecución de una actividad colaborativa del proceso.

Generalización BackBoxProduct

Atributos

- *date:String* La fecha del producto colaborativo que especifica la última vez que fue modificado
- *description:String* Una descripción del producto colaborativo que va es entrada, salida o ambos de una actividad en un proceso colaborativo

- *name:String* (de *BackBoxProduct*) El nombre del elemento del producto colaborativo
- *status:int* Representa el estado como un mecanismo de sincronización de información entre los colaboradores
- *version:String*) La versión del producto colaborativo

Asociaciones

- *includes:BackBoxProduct[0..1]* En la vista individual un producto público podrá estar relacionada con un producto individual

Restricciones Todo producto colaborativo debe ser entrada y/o salida de una actividad colaborativa, esto es, la relación de *entry* o *exit* del mismo debe estar asociada con una *Activity*.

4.2.18. Stakeholder

Descripción La metaclase *Stakeholder* representa cualquier actor, ya sea una persona o una máquina, que se encuentra involucrado en la ejecución de una actividad. Hemos querido establecer la diferenciación entre un actor responsable de una actividad y aquel que participa en la misma, pero que necesita la aprobación del responsable para considerar que dicha actividad ha sido ejecutada correctamente.

Generalización Ninguna

Atributos

- *name:String* El nombre del actor involucrado en el proceso

Asociaciones

- *participates:Activity[0..*]* Conjunto de actividades en el que el actor participa
- *isResponsible:Activity[0..*]* Conjunto de actividades en el que el actor es el responsable de su correcta ejecución

Restricciones Ninguna

4.3. Método para definir el proceso colaborativo

Una vez que ha sido descrito el metamodelo que extiende la propuesta INROMA para poder describir procesos colaborativos, en esta sección vamos a detallar el método para poder hacer uso del mismo.

El objetivo de este método es, a partir de la vista individual de un proceso modelado con la extensión de INROMA, definir el proceso colaborativo. Para ello, es preciso separar la vista individual de un proceso de una organización de la vista colaborativa, manteniendo por tanto la privacidad y la autonomía en la toma de decisiones y compartiendo aquellos elementos que cada organización quiera para poder conformar el proceso en colaboración.

Como si de un proceso se tratara, enumeraremos los pasos a dar a nivel teórico, ya que para facilitar su adopción parte de estas acciones estarán soportadas de forma automática por una herramienta tal y como describiremos en el capítulo 7. La figura 4.3 presenta una vista del método.

En primer lugar hablaremos de las precondiciones, esto es, las condiciones que han de satisfacerse justo antes del comienzo del método, y que son las siguientes:

- Precondición 1 (PRE1): tenemos un escenario de proceso colaborativo a partir de un conjunto de procesos individuales de diversas organizaciones.
- Precondición 2 (PRE2): disponemos de un modelo INROMA para cada uno de los procesos individuales de las organizaciones.

A partir de las precondiciones, el método queda definido por una secuencia de acciones, que gráficamente mostramos en la figura 4.3 y que detallamos a continuación:

- Actividad 1 (A1): deberemos identificar cada uno de los procesos individuales que forman parte de la colaboración y etiquetarlos a su organización correspondiente, generando por tanto las instancias de las *Organization* pertinentes.
- Actividad 2 (A2): tomando como base la influencia del paradigma basado en servicios, cada organización define qué elementos de información consumida (entrada), producida (salida) o modificada (entrada y salida) desea recibir o compartir en la vista en colaboración. De esta forma se identificarán los *Product* que formarán parte del proceso colaborativo y aquellos *BlackBoxProduct* que se mantendrán de forma privada en cada organización.
- Actividad 3 (A3): a partir de los productos en colaboración anteriormente identificados y, tomando como base las actividades descritas a nivel individual por cada organización, será preciso identificar aquellas actividades que se realizan en colaboración. Para garantizar la trazabilidad con los modelos de proceso individuales, todas aquellas actividades que se mantengan de forma autónoma e individual pasarán a ser *BlackBoxActivity* en los modelos individuales.

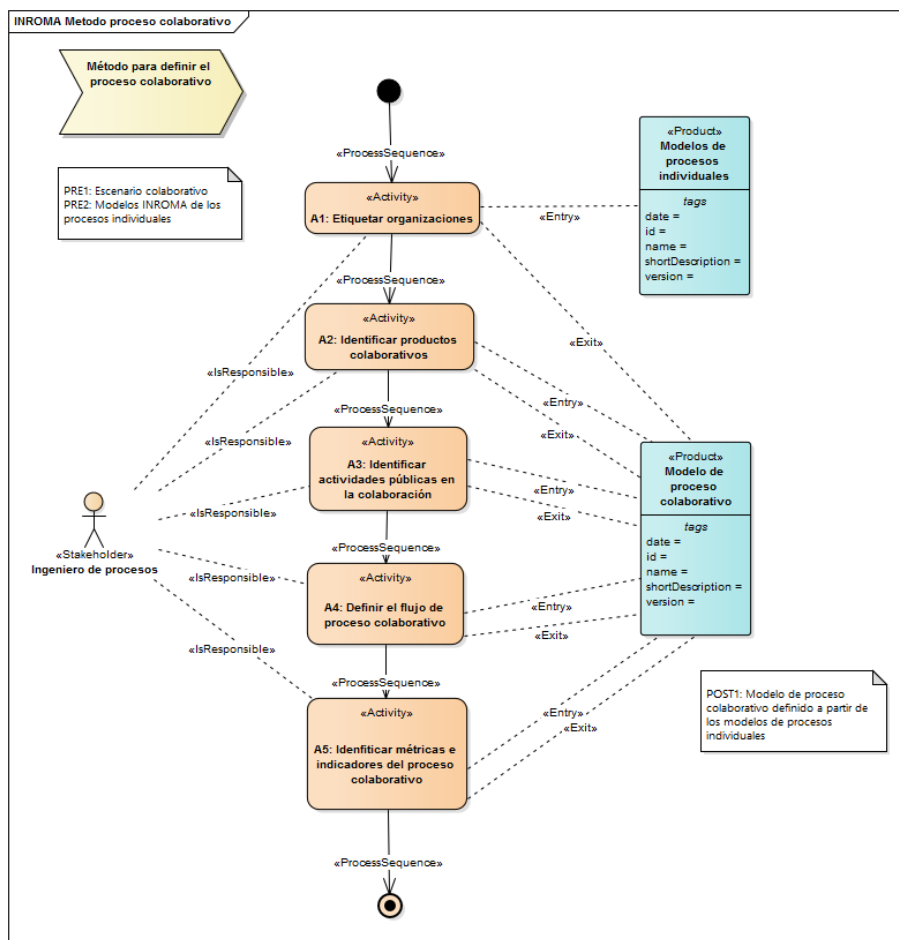


Figura 4.3: Método para definir el proceso colaborativo

- Actividad 4 (A4): una vez que disponemos de los elementos fundamentales de la colaboración, construimos el flujo de las actividades colaborativas siguiendo los elementos de flujo del metamodelo, anotando los participantes y los productos resultado de las mismas.
- Actividad 5 (A5): por último, estableceremos aquellas métricas e indicadores que nos permitan analizar cuantitativamente el proceso en colaboración, elementos que serán anotados como *Metric* e *Indicator* respectivamente. Será posible que las métricas e indicadores coincidan con las individuales, sean un conjunto de las anteriores o sean nuevas, si bien lo habitual será que la forma en la que un proceso se mide a nivel individual no coincida con la vista agregada de una colaboración, por lo que el último escenario será el más habitual. La manera en la que una organización mide y evalúa el proceso a nivel individual se mantendrá como *BlackBoxMetric* y *BlackBoxIndicator*.

Por último, como resultado del método, tendremos la postcondición:

- Postcondición 1 (POST1): modelo de proceso colaborativo definido a partir de los procesos individuales de las organizaciones.

4.4. Conclusiones

En este capítulo se han detallado los primeros componentes del marco de referencia Collaborative Business Generation (CBG), el metamodelo para dar soporte al modelado de procesos de negocio colaborativos y el método para definir el proceso colaborativo.

En primer lugar, hemos analizado INROMA como propuesta de base sobre la que construir nuestro metamodelo para cubrir el enfoque bottom-up para modelar procesos colaborativos. INROMA fue diseñado como un lenguaje auxiliar y por lo tanto consta de los conceptos fundamentales que describen un proceso. El hecho de que INROMA disponga de un método para, a partir de un lenguaje de modelado de procesos, generar de forma automática el modelo INROMA equivalente, hace que ese primer paso bottom-up desde múltiples lenguajes de modelado de procesos quede cubierto, motivo que justifica la elección de esta propuesta como base. Sin embargo, su versión actual no da soporte para expresar los puntos relativos a la colaboración entre los mismos, como la vista multi organizacional y el modelado de actividades y productos en colaboración o la forma de analizar mediante métricas e indicadores dicho proceso colaborativo. Por lo tanto nuestra propuesta extiende INROMA para poder dar soporte al modelado de procesos colaborativos, y en este capítulo hemos detallado los nuevos elementos y presentado el nuevo metamodelo.

Por último, de cara a utilizar la propuesta se ha definido un método para, a partir de un conjunto de procesos individuales descritos con INROMA, generar un proceso colaborativo utilizando el nuevo metamodelo. Este método permite separar la vista individual de un proceso de una organización de la vista colaborativa, manteniendo por tanto la privacidad y la autonomía en la toma de decisiones y compartiendo aquellos elementos que cada organización quiera para poder conformar el proceso en colaboración. Dicho método ha quedado enunciado con sus precondiciones, acciones y postcondiciones.

Capítulo 5

Lenguaje y método para el modelado de casos colaborativos

En el capítulo anterior se han presentado los elementos del marco CBG que dan soporte al modelado de procesos colaborativos, tanto el metamodelo *CBGProcess* para dar soporte al modelado de procesos de negocio colaborativos como el método para, a partir de la vista individual de un proceso modelado conforme a *CBGProcess*, definir el proceso colaborativo. En este capítulo nos centraremos en lo relativo a los casos colaborativos, atendiendo a nuevos componentes de nuestro marco CBG:

- Un metamodelo que, siguiendo el paradigma de gestión de casos, soporta el modelado de casos en entornos colaborativos.
- Un método para, a partir del modelo de casos colaborativos generado, especificar qué elementos forman parte de la vista estática y cuáles en la vista dinámica del caso. De esta forma dispondremos del modelo de caso colaborativo generado a partir del proceso colaborativo.

La primera sección identifica los objetivos que perseguimos para modelar casos colaborativos y que justifican la necesidad de crear un nuevo metamodelo. La segunda sección detalla la propuesta de lenguaje para representar casos colaborativos. En la tercera sección se concreta el método para, atendiendo al metamodelo anterior, definir casos colaborativos. La última sección resume las principales conclusiones del capítulo.

5.1. Lenguajes para modelar casos colaborativos

En el capítulo 3 expusimos que el contexto del problema que pretendemos resolver, la creación de procesos colaborativos a partir de los modelos de procesos individuales de cada organización, coincidía con algunas de las necesidades que se abordan desde la gestión de los casos, en particular lo relativo a poder representar en los modelos de procesos aspectos de planificación dinámica de los flujos, de manera que el proceso en colaboración pueda estar dirigido por los datos y las interacciones y no tanto por las secuencias de flujo.

Dentro de esta sección vamos a enumerar los principales objetivos de diseño del lenguaje para modelar casos colaborativos y justificaremos la necesidad de crear una nueva propuesta que, siendo compatible con las existentes, facilite su adopción práctica en la industria.

Los principales objetivos de diseño para modelar casos colaborativos son:

- **Facilidad en su aprendizaje y uso en la práctica:** dado que uno de los objetivos que nos hemos planteado es tener una validación de la propuesta en la industria, es necesario que el lenguaje contenga los mínimos elementos imprescindibles para poder representar los casos en colaboración, de forma que a partir del uso se establezca el aprendizaje
- **Extensibilidad:** Dado que su sencillez puede hacer que en determinados escenarios se requieran de nuevos elementos, es preciso que el lenguaje disponga de mecanismos que permitan su extensión. Para ello, se plantea la definición del metamodelo atendiendo a UML [OMG, 2011b] y MOF [OMG, 2013].
- **Orientación a gestión de casos:** atendiendo al paradigma de gestión de casos introducidos por Van der Aalst et al. en 2005 [Van der Aalst *et al.*, 2005] para dar soporte a la flexibilidad y al conocimiento intensivo en los procesos de negocio, el lenguaje deberá contener los principales conceptos representados en el metamodelo mostrado en la figura 1.3, en particular lo relativo a un caso, los roles, actividades y datos del caso.
- **Adaptación dinámica:** tomando los principios de la gestión de casos adaptativos, el lenguaje deberá soportar el modelado de un caso de flexible, tomando decisiones en función del entorno y del contexto para seleccionar las alternativas de de alguna forma se han predeterminado. Por ello, el modelo del caso colaborativo contará con una parte estática, a definir en tiempo de diseño, y una dinámica que se concretará en la fase de planificación y ejecución.
- **Soporte a la creación a partir de procesos colaborativos:** dado que nuestro enfoque está orientado a la reutilización de los modelos de procesos individuales siguiendo un enfoque bottom-up, el lenguaje deberá estar orientado para facilitar la creación de modelos a partir de transformaciones M2M desde la extensión del metamodelo INROMA expuesta en el capítulo 4.
- **Relación con estándares existentes:** por último, el lenguaje deberá estar basado en estándares de uso habitual, como UML, y favorecer la conversión a nuevos estándares, como CMMN, mediante transformaciones M2M.

Atendiendo a estos objetivos de diseño, hemos valorado las diversas alternativas existentes, en particular la adopción de CMMN y su extensión para dar soporte a casos colaborativos, que hemos rechazado en favor de crear una nueva propuesta porque:

1. Aunque es menos complejo que otras propuestas de lenguajes de modelado [Marin *et al.*, 2015], sus más de 40 elementos dificultan su adopción en la práctica. Extenderlo para incluir los aspectos inter organizacionales no haría sino aumentar su curva de aprendizaje y dificultad de adopción.
2. Es una propuesta de reciente creación, no hay herramientas para su adopción en la práctica (algunas como Camunda [services GmbH, 2016] han empezado a soportar parte de sus elementos) y los casos piloto se orientan a estudiar su viabilidad en la industria [Miniet y Navarro, 2014] [Herzberg *et al.*, 2014] [Marin y Brown, 2015]. Dado que está respaldado por el OMG, se convertirá en un estándar si bien es posible que evolucione, como evidencia que desde Marzo 2016 se esté trabajando en la versión 1.1 [OMG, 2016].
3. Pese a que dispone de una fase de diseño y otra de planificación y ejecución diferenciadas, no da soporte a todos los elementos y conceptos del paradigma ACM [Kurz *et al.*, 2015], en particular por ser una notación para diseñar casos y no ofrecer guías de cómo representar la fase de ejecución del caso.

Una vez se han establecido tanto sus objetivos de diseño y justificada la elección de crear un nuevo lenguaje para representar casos en colaboración, en la siguiente sección se presenta el metamodelo que constituye la sintaxis abstracta para definir casos colaborativos, contribución de este trabajo de tesis.


```

1 context Case
2 inv multipleEnds : (self.caseElement → select(os|isTypeOf(Final))) → size()
   >= 1

```

Restricción 5.2: Restricción sobre la metaclase *Case*: El caso tiene al menos un elemento final

- Son casos que tienen al menos una tarea El fragmento OCL 5.3 describe la restricción.

```

1 context Case
2 inv atLeastOneTask : (self.caseElement → select(os|isTypeOf(CaseTask))) →
   size() >= 1

```

Restricción 5.3: Restricción sobre la metaclase *Case*: El proceso tiene al menos una tarea

A continuación ofrecemos una explicación detallada de cada una de las metaclases que lo constituyen en orden alfabético. Para cada una de ellas se presenta su descripción, atributos, asociaciones, y sus generalizaciones directas y restricciones si existieran, atendiendo a la misma definición que el propio UML.

5.2.1. Case

Descripción La metaclase *Case* es la clase principal del metamodelo, representando las acciones que serán ejecutadas en base a una planificación dinámica.

Generalización Ninguna

Atributos

- *name:String* El nombre del caso.
- *description:String* Ofrece una descripción del caso de forma textual, incluyendo aspectos tales como la finalidad del mismo o los objetivos que debe cumplir.

Asociaciones

- *organizations:Organization[1..*]* Establece las organizaciones que forman parte del caso
- *elements:CaseElement[3..*] ordered* Conjunto ordenado de los elementos que forman parte de la estructura de un caso
- *sequence:CaseSequence[1..*]* Conjunto de conexiones que muestran el flujo entre los elementos del caso

Restricciones Ninguna

5.2.2. CaseData

Descripción La metaclass *CaseData* representa cualquier pieza de información consumida (entrada), producida (salida) o modificada (entrada y salida) durante la ejecución de una tarea del caso.

Generalización Ninguna

Atributos

- *date:String* La fecha del producto que especifica la última vez que fue modificado
- *description:String* Una descripción del producto que va es entrada, salida o ambos de una actividad en un proceso colaborativo
- *name:String* El nombre del producto del caso
- *status:int* Representa el estado como un mecanismo de sincronización de información entre los colaboradores
- *version:String*) La versión del producto
- *isCollaborative:boolean*) Si el elemento de datos del caso es o no colaborativo

Asociaciones

- *entry:CaseTask[0..*]* Un producto puede ser un elemento de entrada para una o varias tareas del caso
- *exit:CaseTask[0..*]* Un producto puede ser un elemento de salida para una o varias tareas del caso

Restricciones Ninguna

5.2.3. CaseElement

Descripción La metaclass *CaseElement* representa los principales elementos que conforman la estructura de un caso, es decir, los pasos a seguir para la ejecución del mismo.

Generalización Ninguna

Atributos

- *name:String* El nombre del elemento del caso

Asociaciones

- *elements:Case[1]* Establece el caso del que forma parte este elemento
- *incoming:CaseSequence[0..*]* Conexión que tiene el elemento de caso como destino
- *outgoing:CaseSequence[0..*]* Conexión que tiene el elemento de caso como origen

Restricciones Ninguna

5.2.4. CaseFlowElement

Descripción La metaclase *CaseFlowElement* representa los elementos que forman parte de la estructura de un caso pero no se corresponden a la ejecución de ningún fragmento de trabajo.

Generalización *CaseElement*

Atributos

- *name:String (de CaseElement)* El nombre del elemento de flujo

Asociaciones Ninguna

Restricciones Ninguna

5.2.5. CaseSequence

Descripción La metaclase *CaseSequence* establece las conexiones entre dos elementos de caso.

Generalización Ninguna

Atributos

- *name:String* El nombre de la secuencia del caso

Asociaciones

- *sequence:Case[1]* Establece el caso del que forma parte esta conexión de secuencia

- *source:CaseElement[1]* Elemento de caso desde el que parte la conexión
- *target:CaseElement[1]* Elemento de caso hacia el que se dirige la conexión

Restricciones Ninguna

5.2.6. CaseTask

Descripción La metaclase *CaseTask* describe cualquier fragmento de trabajo a ser ejecutado durante el caso colaborativo. A partir de un proceso colaborativo, esta metaclase hará referencia a los elementos tanto públicos como privados de la colaboración, atendiendo al valor de su atributo *isCollaborative*.

Generalización CaseElement

Atributos

- *name:String (de CaseElement)* El nombre de la tarea
- *isCollaborative:boolean* Representa si la tarea del caso es individual o colaborativa

Asociaciones

- *includes:CaseTask[0..*]* Una tarea de un caso puede incluir otras tareas de forma recursiva
- *entry:CaseData[0..*]* Conjunto de productos de entrada para la ejecución de esta tarea
- *exit:CaseData[1..*]* Conjunto de productos de salida que se obtienen como resultado de la ejecución de esta tarea
- *generates:Constraint[0..*]* Conjunto de eventos y restricciones que la tarea genera, como mecanismo de sincronización con otras tareas que aguarden su cumplimiento
- *guards:Constraint[0..*]* Conjunto de eventos y restricciones que la tarea aguarda, como mecanismo de sincronización con otras tareas que las generen

Restricciones Ninguna

5.2.7. Constraint

Descripción La metaclase *Constraint* permite establecer restricciones y condiciones de guarda y eventos como mecanismos de sincronización entre las tareas de un caso.

Generalización Ninguna

Atributos

- *name:String* (de *CaseElement*) El nombre del elemento restricción

Asociaciones

- *guardsConstraint:CaseTask[0..*]* Conjunto de tareas sobre las que la restricción es una guarda (las tareas lo esperan)
- *generatedConstraint:CaseTask[0..*]* Conjunto de tareas sobre las que la restricción es un evento (las tareas lo generan)

Restricciones La restricción que se establece sobre un elemento *Constraint* es que no debe ser guarda y generación de la misma tarea, lo que podría suponer un bloqueo. El fragmento OCL 5.4 describe la restricción.

```

1 context Constraint
2 inv: self.isTypeOf (Constraint) implies forall (guard in self.guardsConstraint |
    guard not in self.generatedConstraint)

```

Restricción 5.4: Restricción sobre la metaclassa *Constraint*: no es guarda y evento de la misma tarea

5.2.8. Conditional

Descripción La metaclassa *Conditional* permite establecer ramas excluyentes en la secuencia lógica de actividades.

Generalización *CaseFlowElement*

Atributos

- *name:String* (de *CaseElement*) El nombre del elemento condicional

Asociaciones Ninguna

Restricciones La restricción que se establece sobre un elemento *Conditional* es que debe tener al menos dos enlaces de salida y, al menos, uno de entrada. El fragmento OCL 5.5 describe la restricción.

```

1 context CaseElement
2 inv: self.isTypeOf (Conditional) implies (self.outgoing → size()>=2 and self.
    incoming → size()>=1)

```

Restricción 5.5: Restricción sobre la metaclassa *Conditional*: al menos, dos enlaces de salida y, al menos, uno de entrada

5.2.9. DiscretionarySequence

Descripción La metaclassa *DiscretionarySequence* establece las conexiones que, con carácter opcional desde el tiempo de diseño, se deben dar entre dos elementos de caso. En tiempo de ejecución se considerará, en el caso de que el caso evolucione hacia dicha secuencia, si las tareas del caso destino de las mismas se deben ejecutar o no.

Generalización CaseSequence

Atributos

- *name:String* (de CaseSequence) El nombre de la secuencia del caso opcional

Asociaciones Ninguna

Restricciones Ninguna

5.2.10. Final

Descripción La metaclassa *Final* representa el punto de salida de un caso, tras el cual podemos decir que se ha finalizado.

Generalización CaseFlowElement

Atributos

- *name:String* (de CaseElement) El nombre del elemento final

Asociaciones Ninguna

Restricciones El elemento final de un caso no puede tener ningún enlace de salida. El fragmento OCL 5.6 describe la restricción.

```

1 context Case
2 inv noOutgoing : (self.caseElement → select(os|isTypeOf(Final))).outgoing → size()
    = 0

```

Restricción 5.6: Restricción sobre la metaclasses *Final*: El elemento final no tiene enlace de salida

5.2.11. HumanTask

Descripción La metaclasses *HumanTask* describe toda tarea del caso desarrollada por un role que representa una persona física.

Generalización *CaseTask*

Atributos

- *name:String* (de *CaseElement*) El nombre de la tarea humana
- *isCollaborative:boolean* (de *CaseTask*) Representa si la tarea humana del caso es individual o colaborativa

Asociaciones

- *participant:Role[0..1]* Rol que participa en la ejecución de una tarea pero no es el responsable de la misma
- *responsible::Role[0..*]* Rol (o roles) responsable de la ejecución de la tarea

Restricciones Ninguna

5.2.12. Initial

Descripción La metaclasses *Initial* representa el punto de entrada al caso, a partir del cual se secuencian el resto de los elementos de caso, pero no corresponde a la ejecución de ningún fragmento de trabajo.

Generalización *CaseFlowElement*

Atributos

- *name:String* (de *CaseElement*) El nombre del elemento de inicio

Asociaciones Ninguna

Restricciones El elemento inicial de un proceso no puede tener ningún enlace de entrada. El fragmento OCL 5.7 describe la restricción.

```

1 context Case
2 inv noIncoming : (self.caseElement → select(os|isTypeOf(Initial))).incoming → size
   () = 0

```

Restricción 5.7: Restricción sobre la metaclass *Initial*: El elemento inicial no tiene enlace de entrada

5.2.13. MandatorySequence

Descripción La metaclass *MandatorySequence* establece las conexiones que, con carácter obligatorio desde el tiempo de diseño, se deben dar entre dos elementos de caso.

Generalización CaseSequence

Atributos

- *name:String* (de *CaseSequence*) El nombre de la secuencia del caso obligatoria

Asociaciones Ninguna

Restricciones Ninguna

5.2.14. Organization

Descripción La metaclass *Organization* representa una organización que forma parte de una CdS sobre la que queremos modelar un caso en colaboración.

Generalización Ninguna

Atributos

- *name:String* El nombre de la organización
- *description:String* Ofrece una descripción de la organización y su rol en la cadena de valor

Asociaciones

- *cases:Case[1..*]* Conjunto de casos de la organización
- *roles:Role[1..*]* Conjunto de roles de la organización

Restricciones Ninguna

5.2.15. Plan

Descripción La metaclase *Plan* permite establecer elementos de flujo entre actividades del caso en los que de forma explícita se representa la capacidad de un rol de planificar de forma dinámica las posibles tareas a ejecutar a continuación.

Generalización *CaseFlowElement*

Atributos

- *name:String (de CaseElement)* El nombre del elemento de planificación

Asociaciones

- *respPlan::Role[0..*]* Rol (o roles) responsable de la secuencia de planificación

Restricciones Ninguna

5.2.16. ProcessTask

Descripción La metaclase *ProcessTask* describe toda tarea del caso que desarrollada por un rol que no representa una persona física.

Generalización *CaseTask*

Atributos

- *name:String (de CaseElement)* El nombre de la tarea de proceso
- *isCollaborative:boolean (de CaseTask)* Representa si la tarea de proceso del caso es individual o colaborativa

Asociaciones Ninguna

Restricciones Ninguna

5.2.17. Role

Descripción La metaclass *Role* representa cualquier actor, ya sea una persona o una máquina, que se encuentra involucrado en la ejecución de una tarea dentro de un caso. Hemos querido establecer la diferenciación entre un actor responsable de una actividad y aquel que participa en la misma, pero que necesita la aprobación del responsable para considerar que dicha actividad ha sido ejecutada correctamente. De igual forma, la actividad de planificación dinámica representada por el elemento de flujo *Plan* tendrá un role responsable de su ejecución.

Generalización Ninguna

Atributos

- *name:String* El nombre del actor involucrado en el caso

Asociaciones

- *participates:HumanTask[0..*]* Conjunto de tareas en el que el actor participa
- *isResponsible:HumanTask[0..*]* Conjunto de tareas en el que el actor es el responsable de su correcta ejecución
- *plans:Plan[0..*]* Conjunto de flujos de planificación dinámica en el que el actor es el responsable

Restricciones Ninguna

5.3. Método para definir casos colaborativos

Una vez que ha sido descrito el metamodelo *CBGCase* poder describir casos colaborativos, en esta sección vamos a detallar el método para poder hacer uso del mismo.

El objetivo de este método es, a partir del modelo de proceso colaborativo conforme a *CBGProcess*, definir el caso colaborativo. Para ello, es preciso incorporar los elementos requeridos para establecer la planificación dinámica de las tareas, esto es:

- Los elementos de secuencia entre tareas que en tiempo de diseño establecemos como obligatorios u opcionales.
- Los elementos de secuencia en los que de forma explícita representamos que se podrá llevar a cabo una replanificación dinámica.
- Los elementos de sincronización a través de restricciones (guardas y eventos) que harán que ciertas tareas entren en la selección de candidatas en esos puntos de replanificación dinámica.

Como si de un proceso se tratara, enumeraremos los pasos a dar a nivel teórico, ya que para facilitar su adopción parte de estas acciones estarán soportadas de forma automática por una herramienta tal y como describiremos en el capítulo 7. La figura 5.2 presenta una vista del método.

En primer lugar hablaremos de las precondiciones, esto es, las condiciones que han de satisfacerse justo antes del comienzo del método, y que son las siguientes:

- Precondición 1 (PRE1): tenemos un modelo de proceso colaborativo conforme a *CBGProcess*.
- Precondición 2 (PRE2): tenemos un escenario en el que se precisa de flexibilidad y dinamismo, requiriendo las capacidades de planificación dinámica que encontramos en el dominio de casos.

A partir de las precondiciones, el método queda definido por una secuencia de acciones, que gráficamente mostramos en el modelo de proceso 5.2 y que enumeramos a continuación:

- Actividad 1 (A1): dado que en el ámbito de los casos los Roles hacen referencia a personas físicas, por defecto todas las tareas derivadas de las actividades del proceso serán *HumanTask*. Por ello deberemos identificar posibles tareas de proceso, asociadas a otros posibles recursos y añadirlas al caso, generando por tanto las instancias de las *ProcessTask* pertinentes.

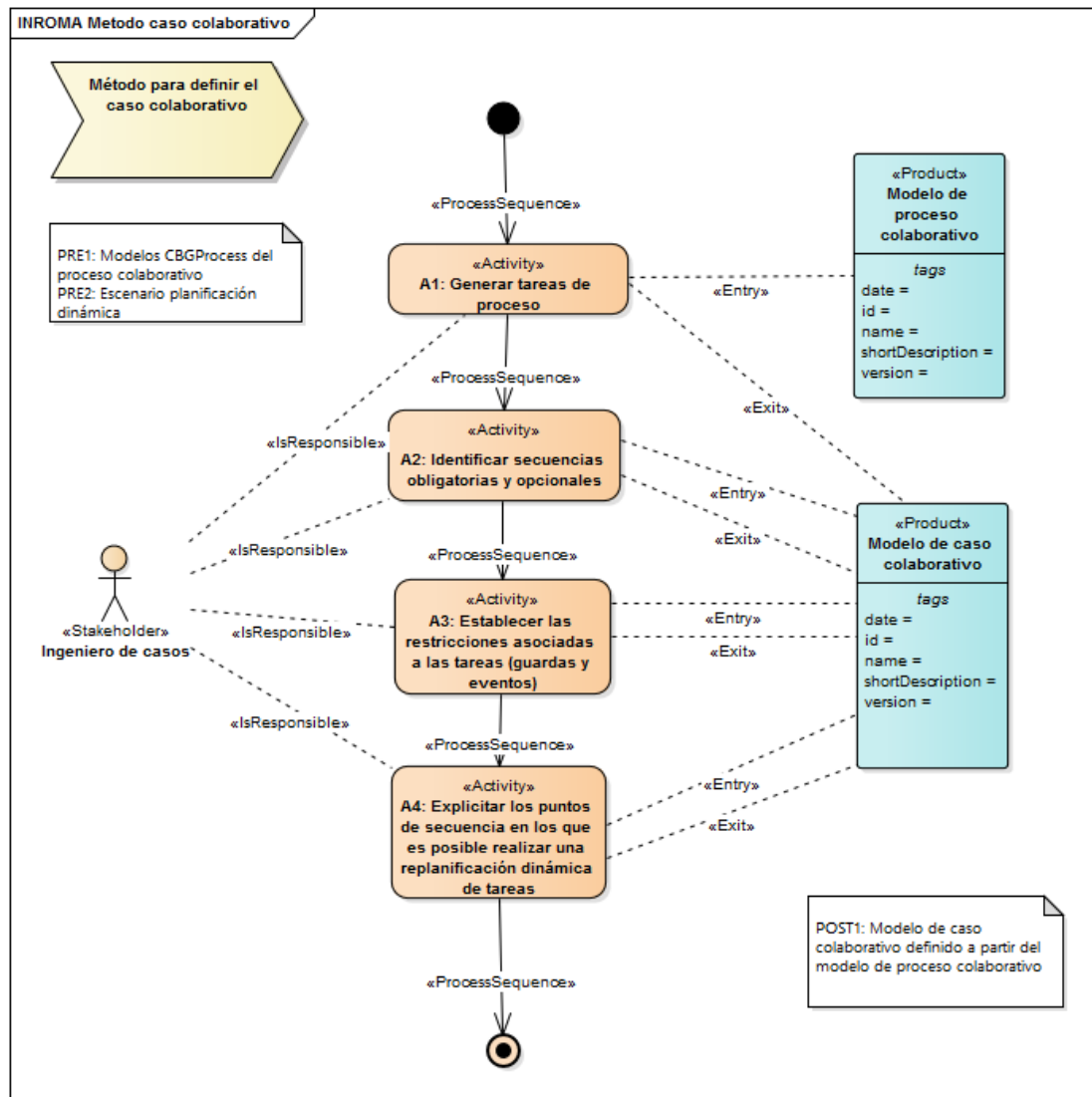


Figura 5.2: Método para definir el caso colaborativo

- Actividad 2 (A2): seguidamente deberemos establecer qué secuencia de elementos del caso es obligatoria y cuál es opcional y se podrá o no llevar a cabo en función de la planificación dinámica. De esta forma concretaremos los elementos de *CaseSequence* definiendo aquellos obligatorios (*MandatorySequence*) y los opcionales (*DiscretionarySequence*).
- Actividad 3 (A3): para poder establecer la sincronización entre tareas, es preciso identificar cuándo una tarea es seleccionable para su ejecución en una replanificación dinámica. Mediante la creación de *Constraint*, podremos identificar cuándo una tarea debe esperar a su cumplimiento (dado que es guardada para que sea elegible) y cuándo la ejecución de una tarea genera el evento que hace que la restricción sea satisfecha.
- Actividad 4 (A4): por último, estableceremos aquellos flujos dentro del caso en los que es posible realizar una planificación dinámica de tareas, creando los *Plan* pertinentes.

Como resultado del método, tendremos la postcondición:

- Postcondición 1 (POST1): modelo de caso colaborativo conforme a *CBGCase* definido a partir del modelo de proceso colaborativo conforme a *CBGProcess*, que a su vez se obtuvo desde los procesos individuales de las organizaciones.

5.4. Conclusiones

En este capítulo se han detallado más componentes del marco CBG, el metamodelo para dar soporte al modelado de casos de negocio colaborativos y el método para definir el caso colaborativo.

En primer lugar hemos definido los objetivos de diseño y justificado la necesidad de crear un nuevo metamodelo y no adoptar CMMN. Tras detallar la propuesta de lenguaje *CBGCase* para representar casos colaborativos, de cara a utilizar la propuesta se ha definido un método para, a partir de un modelo de proceso colaborativo descrito con *CBGProcess*, generar un modelo de caso colaborativo conforme a *CBGCase* que nos permite incluir la capacidad de planificación dinámica, mantenimiento la trazabilidad con el dominio de procesos colaborativos y a su vez con las vistas individuales que cada organización tiene de la colaboración.

Para poder realizar la conversión tanto de las vistas individuales a las colaborativas como del dominio de procesos al de casos vamos a hacer uso de las transformaciones en el ámbito de la ingeniería dirigida por modelos, lo que detallamos en el siguiente capítulo.

Capítulo 6

Transformaciones para la generación de procesos y casos colaborativos

En los capítulos anteriores se han definido los elementos principales del marco de referencia Collaborative Business Generation (CBG) que nos permiten modelar procesos y casos colaborativos con un enfoque bottom-up, en particular:

- Un metamodelo, denominado *CBGProcess*, para dar soporte al modelado de procesos de negocio colaborativos siguiendo un enfoque bottom-up reutilizando por tanto los modelos de procesos individuales.
- Un método para, a partir de la vista individual de un proceso conforme a *CBGProcess*, definir el proceso colaborativo.
- Un metamodelo, denominado *CBGCase*, que siguiendo el paradigma de gestión de casos soporta el modelado de casos en entornos colaborativos.
- Un método para, a partir del modelo de casos colaborativos generado, especificar qué elementos forman parte de la vista estática y cuáles en la vista dinámica del caso.

En este capítulo detallaremos cómo, aplicando las transformaciones en el ámbito de la ingeniería dirigida por modelos, se realiza la conversión tanto de las vistas individuales a las colaborativas como del dominio de procesos al de casos. De esta forma, como se describe en la figura 6.1 quedarán cubiertos otros componentes del marco CGB, en particular:

- Una transformación M2M para generar la vista colaborativa desde un modelo de proceso individual.
- Una transformación M2M que permite, a partir de un proceso colaborativo conforme a *CBGProcess*, generar automáticamente una base de modelo de casos colaborativos conforme a *CBGCase*. De esta forma se consigue añadir la componente dinámica a los procesos, realizando una conversión del dominio de procesos al dominio de casos.

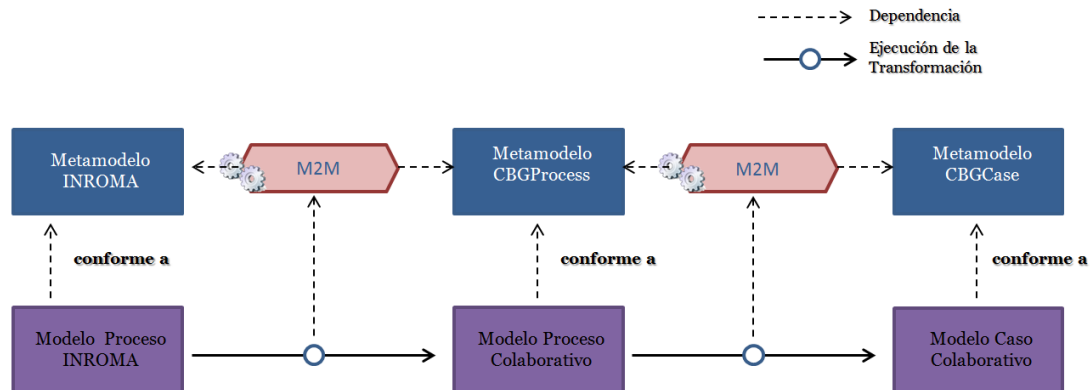


Figura 6.1: Transformaciones M2M en el marco CBG

Con este fin el capítulo se estructura como sigue. En la primera sección se detalla cómo vamos a hacer uso de las transformaciones en el ámbito de la ingeniería del software dirigida por modelos, ampliando la influencia del paradigma basado en modelos que hemos enunciado en la sección 3.4.2 cuando describimos el planteamiento del problema en el capítulo 3. Seguidamente se exponen las transformaciones que forman parte del marco CBG. Por último, se presenta un resumen con las principales conclusiones del capítulo.

6.1. Selección de QVT para describir las transformaciones entre modelos

En MDE todavía no existe una base establecida sobre la que podamos describir las transformaciones M2M [Di Ruscio *et al.*, 2012], siendo muchas las propuestas para llevarlas a cabo y diferentes los criterios para la elección de una de estas propuestas frente a otra. Por ello, en la última década se han planteado diversas soluciones aplicando diversos paradigmas, enfoques de modelado o soporte de herramientas, clasificándolas en base a su enfoque en [Czarnecki y Helsen, 2006]:

- *Basados en reglas.* Utilizan lenguajes de reglas descritas como *guarda - acción*.
- *Basado en manipulación directa.* A través de una interfaz de programación se accede a la representación interna de los modelos y se desarrolla la transformación, generalmente en un entorno orientado a objetos.
- *Basado en transformación de grafos.* La descripción de los modelos de origen y destino se realiza atendiendo a las soluciones y marcos derivados de la teoría de grafos. En este grupo se incluyen AGG [Taentzer, 2004], TGG [Schürr, 1995] o VIATRA2 [Varró *et al.*, 2002].
- *Operacional.* Similar a la manipulación directa pero con una interfaz dedicada para realizar las transformaciones sobre los modelos. En este grupo se incluyen QVT Operational mappings [OMG, 2011a] o Kermeta [Muller *et al.*, 2005].
- *Relacional.* Utiliza relaciones matemáticas y mediante lenguajes declarativos se resuelven las restricciones dadas para ejecutar la transformación. En este enfoque las propuestas no tienen efectos laterales, a diferencia de las transformaciones con enfoque operacional. Adicionalmente, estas propuestas dan soporte de forma natural a la multidireccionalidad de las reglas (Modelo A a Modelo B y viceversa). En este grupo se incluyen QVT Relations [OMG, 2011a] o AMW [Del Fabro *et al.*, 2005].
- *Híbrido.* Las propuestas de este tipo combinan diferentes técnicas operacionales y relacionales. A este grupo pertenecen ATL [Jouault *et al.*, 2008] [Jouault y Kurtev, 2006] y ETL [Kolovos *et al.*, 2008], los cuales envuelven cuerpos imperativos en sentencias declarativas.

A la vista de esta clasificación podemos ver que el número de propuestas para llevar a cabo las transformaciones es diversa. En nuestro trabajo hemos apostado por QVT a la hora de definir las transformaciones del marco CBG, por los siguientes motivos:

- Estándar del OMG: aunque en su versión actual existen algunos aspectos sin resolver y no cuenta con el soporte de herramientas, al ser el estándar propuesto por OMG esperamos que en el futuro estas posibles debilidades sean corregidas.
- Bidireccionalidad: QVT Relations nos la ofrece, siendo una característica necesaria para establecer el paso de modelo a caso o viceversa dentro del marco CGB.

- Representación gráfica: entendemos que la vista gráfica de QVT puede facilitar su comprensión a personas ajenas al campo de la programación.
- Experiencia: dentro del grupo de Ingeniería Web y Testing Temprano (IWT2) en el que se desarrolla este trabajo, existe experiencia en el uso de QVT frente al resto de las propuestas

Todos estos elementos han propiciado la elección de utilizar QVT Relations para las transformaciones del marco CBG, que se muestran a continuación.

6.2. Transformaciones en el marco CBG

En esta sección vamos a presentar la forma en la que se han llevado a la práctica las transformaciones dentro del marco CBG. De la misma forma que especificamos los métodos, mediante un modelo INROMA, presentamos el proceso general de cómo se lleva a cabo esta tarea como se recoge en la figura 6.2. En ella se observa cómo a partir de las correspondencias entre los metamodelos se definen las transformaciones pertinentes en QVT Relations mediante la actividad *Generar transformaciones M2M*.

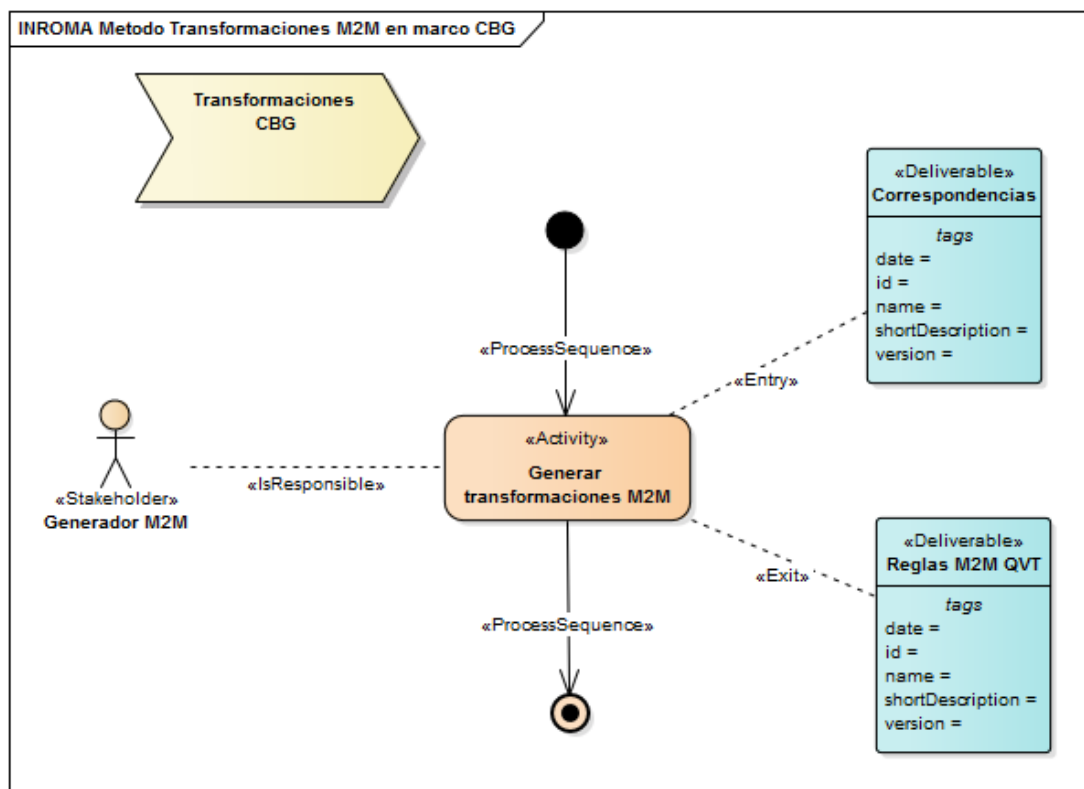


Figura 6.2: Proceso de generación de transformaciones en el marco CBG

Seguidamente se detalla cómo se ha llevado a cabo el proceso anterior para cada una de las transformaciones incluidas en el marco CGB.

6.2.1. Transformación M2M horizontal para generar la vista colaborativa de un proceso individual

Atendiendo al método general del marco CBG, para poder construir el modelo de proceso en colaboración a partir de los modelos de procesos individuales, es preciso crear una vista colaborativa de los mismos. Para ello, hemos realizado una extensión del metamodelo INROMA para dar soporte a las nuevas necesidades.

El nuevo metamodelo está al mismo nivel de abstracción que INROMA, por lo que en este caso estaremos realizando una transformación M2M horizontal, atendiendo a la descripción realizada en la sección anterior.

En el contexto de las transformaciones M2M en el marco CBG que hemos mostrado en la figura 6.1, esta transformación horizontal queda detallada en la figura 6.3.

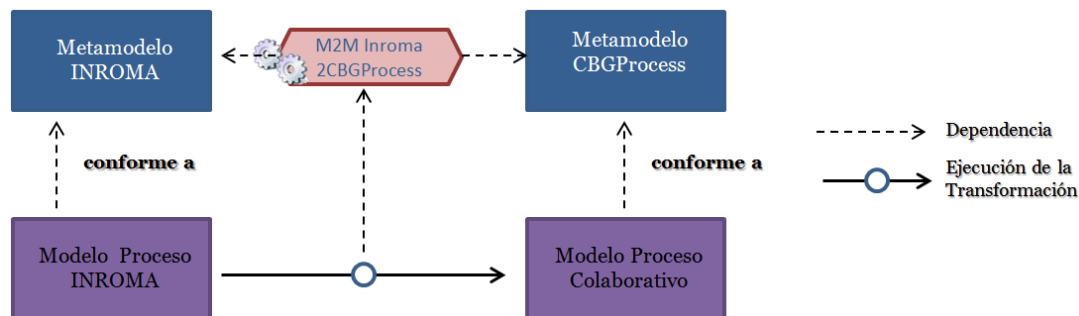


Figura 6.3: Transformación M2M INROMA2CBGProcess en el marco CBG

Utilizando la descripción de QVT, en la figura 6.4 se observa la relación principal entre el *Process* de INROMA y su equivalente en CBGProcess. En este caso deberemos generar por defecto al menos una *Organization*, elemento presente únicamente en la extensión del metamodelo INROMA. De forma recursiva, se deben ir generando las relaciones entre los elementos derivados de un *Process*, en particular:

- El elemento *Activity*, a través de la relación *MapActivities*.
- El elemento *Indicator*, a través de la relación *MapIndicators*.
- El elemento *Initial*, a través de la relación *MapInitial*.
- El elemento *Final*, a través de la relación *MapFinal*.
- El elemento *Conditional*, a través de la relación *MapConditional*.

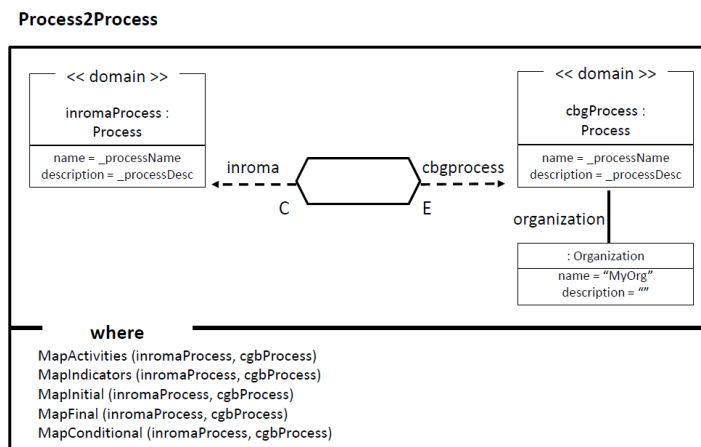


Figura 6.4: Vista QVT Relations de la relación Process2Process en la transformación M2M inroma2CBGProcess

El resto de transformaciones M2M entre INROMA y CBGProcess se encuentran detalladas en el Anexo A, siguiendo una correspondencias entre las relaciones y los diagramas de transformación QVT tal y como se resume en la tabla 6.1.

Relación	Sintaxis Gráfica QVT Relacional
Process2Process	Figura A.1
MapActivities	Figura A.2
MapIndicators	Figura A.3
MapInitial	Figura A.4
MapFinal	Figura A.5
MapConditional	Figura A.6
MapInputProductsOfActivity	Figura A.7
MapOutputProductsOfActivity	Figura A.8
MapStakeholdersPartOfActivity	Figura A.9
MapStakeholdersRespOfActivity	Figura A.10
MapIncomingSequence	Figura A.11
MapOutgoingSequence	Figura A.12
MapMetrics	Figura A.13

Tabla 6.1: Relaciones para acometer la transformación INROMA2CBGProcess

6.2.2. Transformación M2M de procesos a casos colaborativos

De igual forma, para poder pasar del dominio de procesos al de casos, utilizaremos una transformación M2M que permitirá obtener modelos de casos conforme al metamodelo CBGCase, a partir de modelos de procesos colaborativos conformes al metamodelo CBGProcess, tal y como se muestra en la figura 6.5.

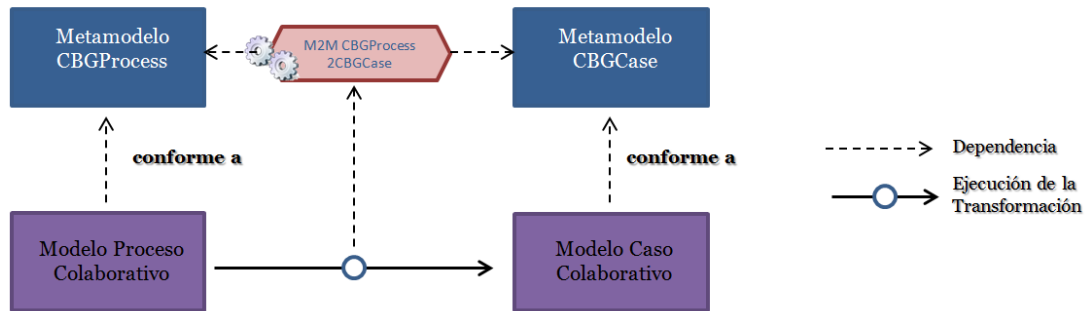


Figura 6.5: Transformación M2M CBGProcess2CBGCase en el marco CBG

Process2Case

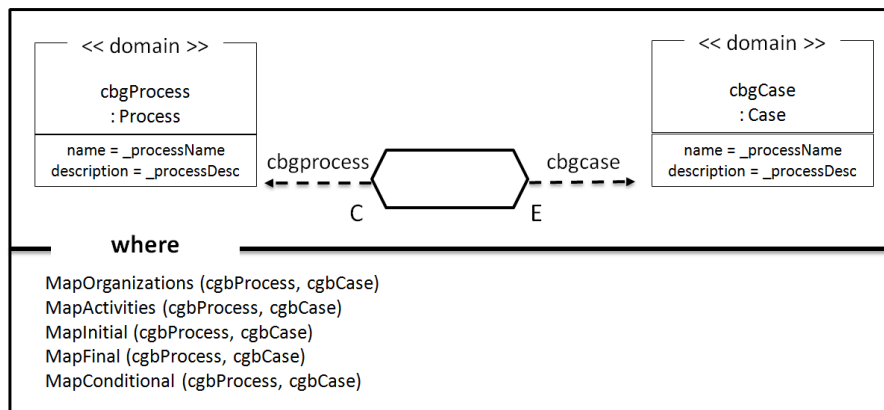


Figura 6.6: Vista QVT Relations de la relación Process2Case en la transformación M2M CBG-Process2CBGCase

Así, en la figura 6.6 se observa la relación principal entre el *Process* de CBGProcess y su equivalente como *Case* en CBGCase. De forma recursiva, la transformación va generando las relaciones entre los elementos derivados de un *Process*, en particular:

- El elemento *Organization*, a través de la relación *MapOrganizations*.
- El elemento *Activity*, a través de la relación *MapActivities*.
- El elemento *Initial*, a través de la relación *MapInitial*.

- El elemento *Final*, a través de la relación *MapFinal*.
- El elemento *Conditional*, a través de la relación *MapConditional*.

En este caso, dado que los modelos conformes a CBGProcess pueden contener tanto la vista colaborativa como la trazabilidad individual de un proceso en colaboración, la transformación incluye todos los elementos del segundo caso, en el que es preciso incluir:

- Las actividades individuales que forman parte de un proceso colaborativo, tal y como muestra la figura 6.7 a través de la relación *MapIncludedActivitiesOfActivity*.
- Los productos individuales que forman parte de un proceso colaborativo, tal y como muestra la figura 6.8 a través de la relación *MapInputProductsOfActivity*. De igual forma procedemos con los productos que son resultados de una actividad del proceso.

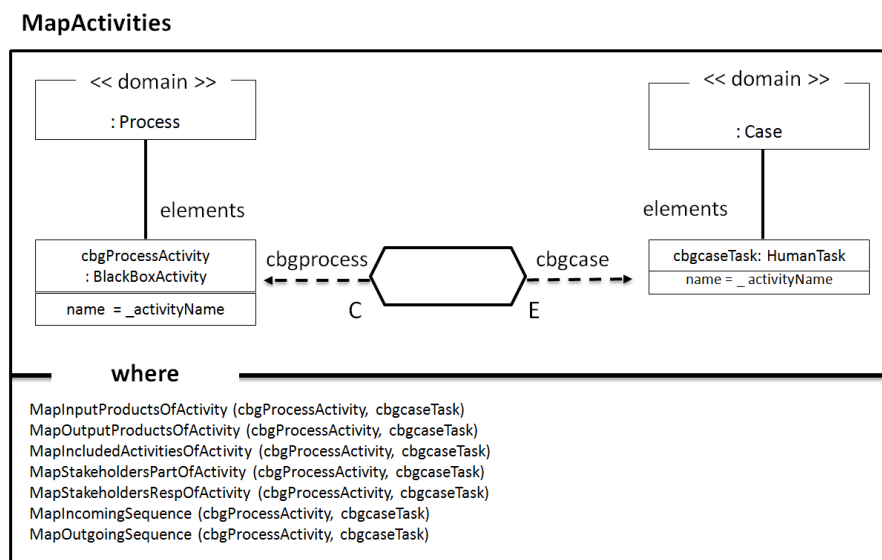


Figura 6.7: Vista QVT Relations de la relación MapActivities en la transformación M2M CBG-Process2CBGCase

El resto de transformaciones M2M entre CBGProcess y CBGCase se encuentran detalladas en el Anexo B, siguiendo una correspondencias entre las relaciones y los diagramas de transformación QVT tal y como se resume en la tabla 6.2.

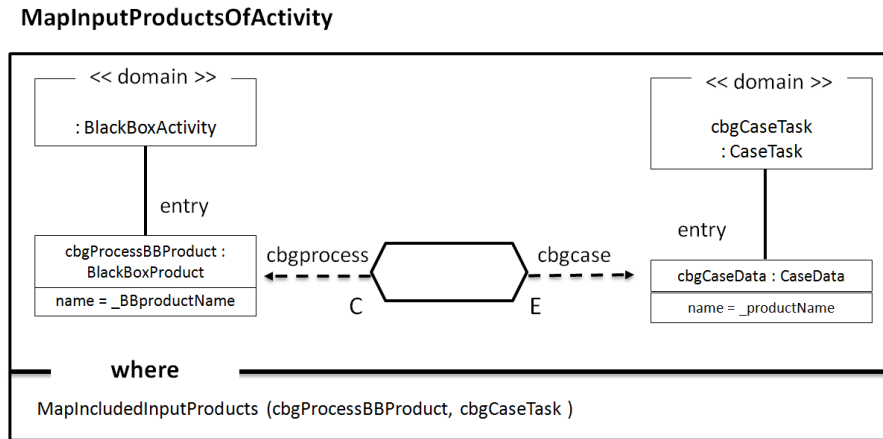


Figura 6.8: Vista QVT Relations de la relación MapInputProductsOfActivity en la transformación M2M CBGProcess2CBGCase

Relación	Sintaxis Gráfica QVT Relacional
Process2Case	Figura B.1
MapOrganizations	Figura B.2
MapActivities	Figura B.3
MapInitial	Figura B.4
MapFinal	Figura B.5
MapConditional	Figura B.6
MapStakeholders	Figura B.7
MapInputProductsOfActivity	Figura B.8
MapOutputProductsOfActivity	Figura B.9
MapIncludedInputProducts	Figura B.10
MapIncludedInputProducts	Figura B.10
MapIncludedOutputProducts	Figura B.11
MapIncludedActivitiesOfActivity	Figura B.12
MapStakeholdersPartOfActivity	Figura B.13
MapStakeholdersRespOfActivity	Figura B.14
MapIncomingSequence	Figura B.15
MapOutgoingSequence	Figura B.16

Tabla 6.2: Relaciones para acometer la transformación CBGProcess2CBGCase

6.3. Conclusiones

En este capítulo hemos detallado de qué forma se aplican las transformaciones del dominio de la ingeniería dirigida por modelos para, en nuestro marco CBG, convertir un proceso colaborativo a un caso. Hemos justificado el uso de QVT como lenguaje para definir las transformaciones desde un punto de vista teórico y hemos presentado su sintaxis gráfica y textual.

Por otro lado hemos definido el proceso para generar nuevas transformaciones en el marco CBG, de forma que se puedan añadir nuevos lenguajes en el futuro, y hemos detallado las dos transformaciones básicas del marco:

- Una transformación M2M para generar la vista colaborativa desde un modelo de proceso individual.
- Una transformación M2M que permite, a partir de un proceso colaborativo conforme a *CBGProcess*, generar automáticamente una base de modelo de casos colaborativos conforme a *CBGCase*.

Debido a la extensión de las relaciones incluidas en las transformaciones no se han incluido en este capítulo en su totalidad, refiriendo al lector a los Anexos A y B donde se detallan las transformaciones entre INROMA y CBGProcess y entre CBGProcess y CBGCase respectivamente.

Una vez hemos presentado los metamodelos *CBGProcess* y *CBGCase*, así como las transformaciones desde un punto de vista teórico haciendo uso de QVT, en el capítulo 7 pasamos a exponer la herramienta CGB-Tool que implementa estas transformaciones y facilita su adopción práctica.

Capítulo 7

CBG-Tool: Herramienta de soporte para la generación de procesos y casos de negocios colaborativos

En los capítulos anteriores se han definido los elementos principales del marco de referencia Collaborative Business Generation (CBG) que nos permiten modelar procesos y casos colaborativos con un enfoque bottom-up, en particular:

- Un metamodelo, denominado *CBGProcess*, para dar soporte al modelado de procesos de negocio colaborativos.
- Una transformación M2M para generar la vista colaborativa desde un modelo de proceso individual.
- Un método para, a partir de la vista individual de un proceso conforme a *CBGProcess*, definir el proceso colaborativo. Dicho método permite separar la vista individual de un proceso de una organización de la vista colaborativa, manteniendo por tanto la privacidad y la autonomía en la toma de decisiones y compartiendo aquellos elementos que cada organización quiera para poder conformar el proceso en colaboración.
- Un metamodelo, denominado *CBGCase*, que siguiendo el paradigma de gestión de casos soporta el modelado de casos en entornos colaborativos.
- Una transformación M2M que permite, a partir de un proceso colaborativo conforme a *CBGProcess*, generar automáticamente una base de modelo de casos colaborativos conforme a *CBGCase*. De esta forma se consigue añadir la componente dinámica a los procesos, realizando una conversión del dominio de procesos al dominio de casos.
- Un método para, a partir del modelo de casos colaborativos generado, especificar qué elementos forman parte de la vista estática y cuáles en la vista dinámica del caso. De esta forma dispondremos del modelo de caso colaborativo generado a partir del proceso colaborativo.

En este capítulo detallaremos el último de los elementos que forman parte del marco CGB, una herramienta de soporte denominada CBG-Tool para facilitar la creación de modelos y la ejecución de las transformaciones desde un punto de vista práctico.

Para conseguir este objetivo el presente capítulo se estructura como sigue. En la primera sección se detallan las funcionalidades a partir de la descripción de sus casos de uso, se presenta la arquitectura de la solución y se expone cuál va a ser el entorno de desarrollo de ejecución de CBG-Tool. Seguidamente se presenta de forma detallada cómo la herramienta da soporte a la edición de modelos y cómo implementa las transformaciones. En la tercera sección se realizará la validación con las situaciones reales que han justificado la realización de este trabajo de tesis y que fueron descritas en los capítulos iniciales, utilizando CBG-Tool como implementación de la solución. Por último, en la última sección se mencionan las conclusiones más relevantes del capítulo.

7.1. Casos de uso, entorno de trabajo y arquitectura de CBG-Tool

En esta sección se detallan las funcionalidades a partir de la descripción de sus casos de uso, se presenta la arquitectura de la solución y se expone cuál va a ser el entorno de desarrollo de ejecución de CBG-Tool.

7.1.1. Casos de uso

En esta sección presentamos las funcionalidades requeridas para la herramienta a partir de la descripción de sus casos de uso, mostrados en la figura 7.1.

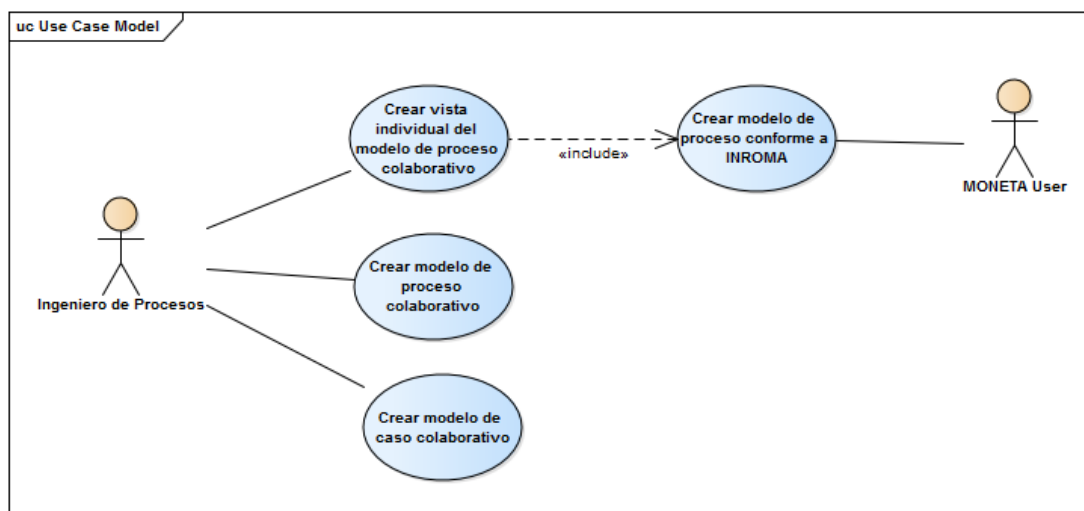


Figura 7.1: Casos de uso de la herramienta CBG-Tool

Como hemos expuesto en los capítulos anteriores, el marco CBG está influenciado por la solución INROMA, que a su vez dispone de una herramienta de soporte denominada *MONETA* y detallada en la tesis de García-Borgoñón[García-Borgoñón, 2016]. Por ello hemos representado al usuario de *MONETA* y al caso de uso asociado que permite, desde una diversidad de lenguajes de modelado de procesos, generar un modelo de proceso conforme al metamodelo INROMA. De esta forma, el primer y último paso del marco global CBG consistirá en el uso de la herramienta *MONETA*, que se establece como precondition para el uso de CBG-Tool para dar soporte a diversos lenguajes de modelado de procesos, y al no formar parte de CBG los ejemplos y validación de los casos de uso se realizarán a partir del modelo INROMA de los procesos, refiriendo al lector a la tesis de García-Borgoñón en la que se detalla ampliamente la funcionalidad *MONETA* complementaria al marco CBG que nos ocupa.

CBG-Tool tendrá un único actor, representado como el *Ingeniero de Procesos*, que será el responsable de crear los modelos de procesos y casos, tanto a nivel individual como en la posible colaboración.

La herramienta da soporte a tres casos de uso que se detallan a continuación, en las tablas 7.1, 7.2 y 7.3:

Caso de uso	Crear vista individual del modelo de proceso colaborativo
Actor	Ingeniero de Procesos
Descripción	A partir de los modelos de procesos individuales (conformes a INROMA), se genera la vista individual del proceso en colaboración
Precondiciones	Disponibilidad del modelo de proceso INROMA de cada participante en la colaboración
Postcondiciones	Vista individual de los procesos en colaboración, detallando los elementos individuales y los colaborativos
Incluye	Crear modelo de procesos conforme a INROMA, utilizando la solución MONETA para dar soporte a modelos de procesos conforme a cualquier lenguaje de modelado de procesos
Flujo Normal	Cada organización dispone de su modelo de proceso individual conforme a INROMA y genera la vista individual de la colaboración, estableciendo aquellos elementos privados y cuáles forman parte del proceso colaborativo
Flujos Alternativos	Si una organización ya participaba en una colaboración y desea participar en otra colaboración, modifica la vista individual anterior incluyendo los nuevos elementos individuales y colaborativos considerando todas las CdS en las que participa

Tabla 7.1: Descripción del caso de uso Crear vista individual del modelo de proceso colaborativo

Caso de uso	Crear modelo de proceso colaborativo
Actor	Ingeniero de Procesos
Descripción	A partir de los elementos públicos (colaborativos) de los modelos de procesos individuales se crea el modelo de proceso colaborativo
Precondiciones	Disponibilidad de la vistas individuales de los modelos de procesos en la colaboración que nos ocupa
Postcondiciones	Modelo de proceso en colaboración
Flujo Normal	Cada organización dispone de su vista individual de la colaboración y, a partir de los elementos públicos (colaborativos) de los mismos, siguiendo el método para definir el proceso colaborativo descrito en la sección 4.3, se crea el modelo de proceso en colaboración
Flujos Alternativos	Si el modelo de proceso colaborativo ya existía pero se integran nuevas organizaciones en la CdS, se modifica el modelo atendiendo a los nuevos posibles elementos colaborativos y Stakeholders

Tabla 7.2: Descripción del caso de uso Crear modelo de proceso colaborativo

Caso de uso	Crear modelo de caso colaborativo
Actor	Ingeniero de Procesos
Descripción	A partir de los modelos de procesos en colaboración, se genera el equivalente en el dominio de casos
Precondiciones	Disponibilidad de los modelos de procesos en colaboración
Postcondiciones	Modelo de caso en colaboración
Flujo Normal	Se dispone del modelo de proceso en colaboración y se genera el equivalente en el dominio de casos
Flujos Alternativos	Si el modelo de caso colaborativo ya existía pero se integran nuevas organizaciones en la CdS, se modifica el modelo atendiendo a los nuevos posibles elementos colaborativos

Tabla 7.3: Descripción del caso de uso Crear modelo de caso colaborativo

7.1.2. Entorno de trabajo de CBG-Tool: Enterprise Architect

Para desarrollar CBG-Tool hemos optado por la herramienta Enterprise Architect (EA) [SparxSystems, 2014] como entorno de ejecución por los siguientes motivos:

- Por ser la herramienta de base en el grupo IWT2, decisión tomada como conclusiones del estudio elaborado para la Consejería de Cultura y Deporte de la Junta de Andalucía [IWT2 y Consejería de Cultura de la Junta de Andalucía, 2013].
- Dado que la propuesta CBG extiende la solución INROMA, que cuenta con la herramienta MONETA, también desarrollada en EA, para mantener el mismo entorno ofreciendo una solución global y compatible con la anterior.
- Para reducir el tiempo de desarrollo respecto a una solución desde cero (*from scratch*), reutilizando las funcionalidades de base en lo relativo a edición de modelos y capacidades de extensión propias de EA.

CGB-Tool se configura como una implementación de referencia como herramienta de soporte al marco CBG, si bien es posible utilizar otros entornos para desarrollar los modelos e implementar de forma práctica las transformaciones expuestas en el capítulo anterior. De esta forma, CGB-Tool nos permite validar la propuesta de una forma práctica y puede facilitar la transferencia de los resultados de este trabajo de tesis a empresas. Adicionalmente hemos realizado pruebas de concepto con otras herramientas de cara a reducir el riesgo de que la propuesta estuviera acoplada a un entorno concreto, algo que hemos evidenciado haciendo uso del IDE Eclipse [The Eclipse Foundation, 2014b] y sus diferentes extensiones para el modelado, metamodelado y definición de transformaciones (Eugenia [The Eclipse Foundation, 2014c], QVTMedini [IKV++, 2011], BPMN Modeler [The Eclipse Foundation, 2014a], Papyrus [The Eclipse Foundation, 2015], etc.) comprobando la viabilidad de su posible elección como herramienta base.

Para poder incluir los nuevos metamodelos dentro de la herramienta, ya hemos visto en capítulos anteriores que un lenguaje de modelado tiene una sintaxis concreta y una sintaxis abstracta, siendo esta última su metamodelo [Brambilla *et al.*, 2012b]. Para definir la sintaxis concreta hemos apostado por extender UML creando un *perfil UML* [Fontoura *et al.*, 2000], de forma que facilitemos la adopción de estos lenguajes al estar basado en un estándar extendido en la industria y favoreciendo la compatibilidad con herramientas basadas en UML. Tal y como se establece en la especificación de UML, la enunciación de un perfil pasa por la definición personalizada de tres componentes:

1. Estereotipos (*Stereotype*), que permiten definir elementos de dominio específico, es decir, los elementos del metamodelo. Una vez estereotipados es necesario establecer las metaclases UML de las que extienden.
2. Restricciones (*Constraints*), mecanismo que permite establecer condiciones sobre los elementos estereotipados del metamodelo que nos permitan verificar un modelo bien formado.
3. Valores etiquetados (*Tagged value*), que nos permiten añadir propiedades adicionales y asociarlas a un elemento estereotipado del metamodelo.

Dado que EA incorpora la funcionalidad de definición de nuevos perfiles UML, será la forma de llevar a la práctica la inclusión de los nuevos metamodelos dentro de CBG-Tool. Para llevar a cabo las transformaciones, hemos utilizado el mecanismo de extensión de componentes, conocido como *Add-In* y representado en la figura 7.2, para implementar las transformaciones en el lenguaje C#.

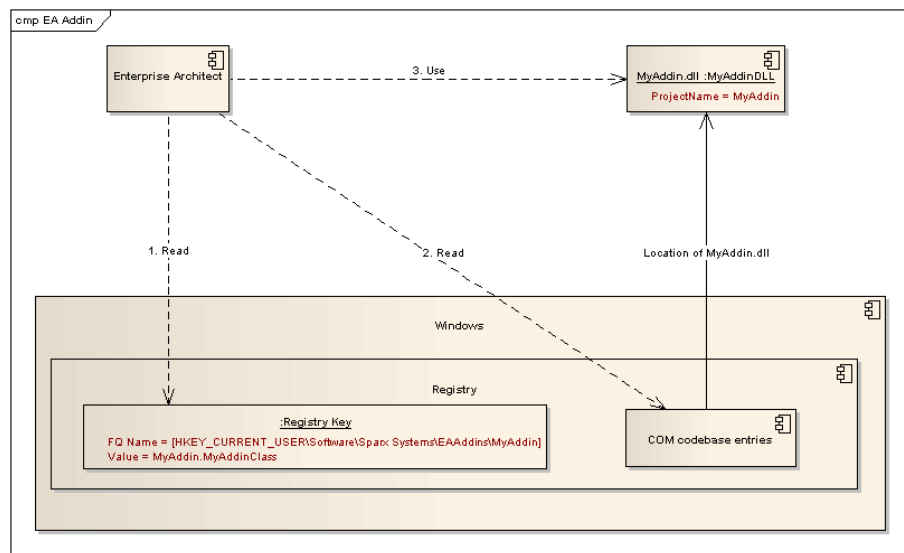


Figura 7.2: Arquitectura de extensión de componentes de Enterprise Architect

7.1.3. Arquitectura

La solución CBG-Tool se constituye de esta forma como un Add-in sobre la herramienta Enterprise Architect, que incluye tanto la implementación de las transformaciones como los perfiles UML de los metamodelos CBGProcess y CBGCase. La figura 7.3 muestra la arquitectura de la herramienta dentro del marco EA.

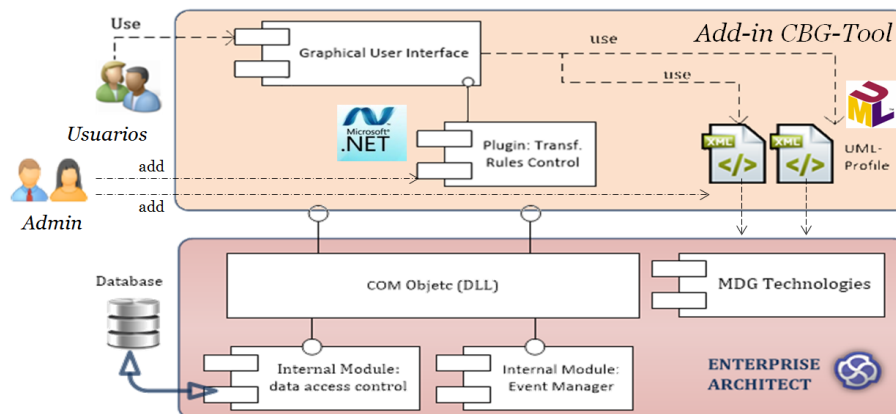


Figura 7.3: Arquitectura de la herramienta CBG-Tool dentro de Enterprise Architect

CBG-Tool da soporte dentro del marco CBG a la edición de modelos conformes a los metamodelos CBGProcess y CBGCase, a través de los perfiles UML, así como a las transformaciones desde INROMA a procesos colaborativos, y desde procesos a casos colaborativos implementadas en el Add-in. La figura 7.4 muestra la arquitectura de la solución.

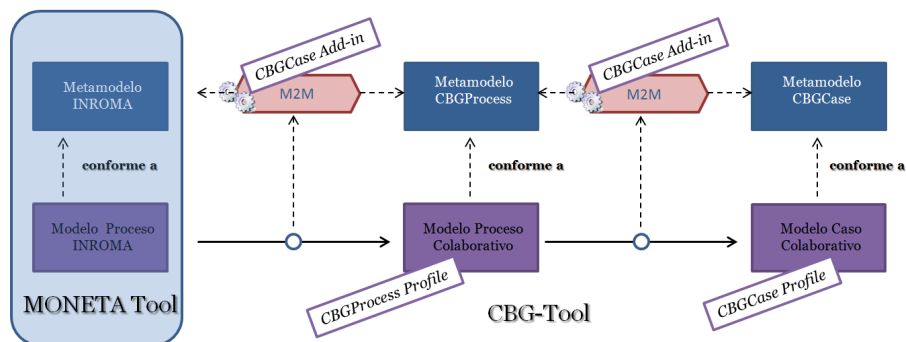


Figura 7.4: Arquitectura de la herramienta CBG-Tool dentro del marco CBG

7.2. La herramienta CBG-Tool

Una vez determinado el entorno de trabajo en el que se desarrolla CBG-Tool, que como ya hemos comentado está basado en Enterprise Architect, en esta sección vamos a detallar cómo la solución da soporte a la edición de modelos y la ejecución de las transformaciones.

7.2.1. El lenguaje de modelado de procesos colaborativos CBGProcess en CBG-Tool

Tal y como hemos visto en la sección anterior, la incorporación de CBGProcess en la herramienta se realiza por medio de la definición de un perfil de UML, el cual se observa en la figura 7.5.

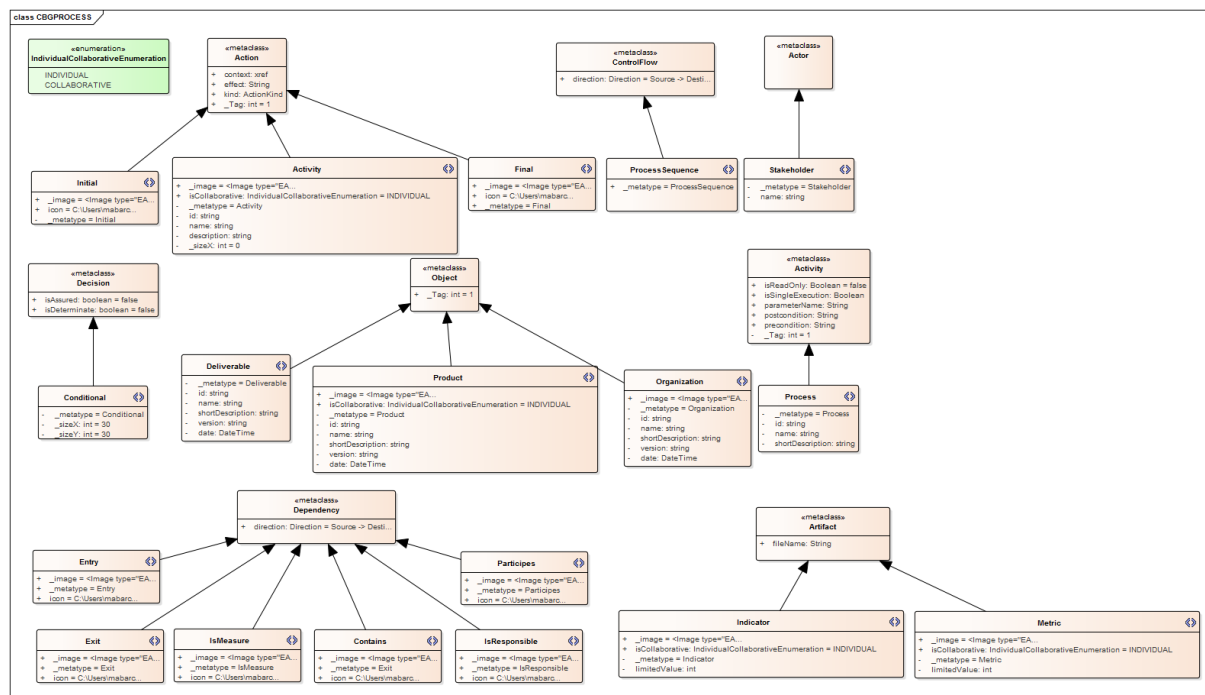


Figura 7.5: Perfil UML para CBGProcess

En la tabla 7.4 se muestran los estereotipos del perfil UML de CBGProcess, así como la metaclass que se ha utilizado para realizarlo, y se justifica la elección de la metaclass correspondiente a cada estereotipo.

Metaclassa UML	Estereotipo	Justificación
Activity	Process	La metaclassa <i>Activity</i> representa la secuencia coordinada de acciones, y por ellos ha sido elegida para ser extendida con <i>Process</i> .
Action	Activity	La metaclassa <i>Action</i> representa un simple paso dentro de una <i>Activity</i> , lo que en CBGProcess se ha denominado <i>Activity</i>
	Initial	
	Final	
Decision	Conditional	Este elemento se corresponde a un <i>ControlNode</i> de UML, más concretamente los que hacen referencia a <i>DecisionNode</i> , que en EA está definido como <i>Decision</i> .
Object	Deliverable	Esta meclase de EA hace referencia a la metaclassa <i>ObjectNode</i> de UML, que se define como un <i>ActivityNode</i> que indica una instancia de un estado de un clasificador concreto, y por eso aquí se ha elegido para representar los <i>Products</i> de CBGProcess.
	Product	
Artifact	Metric	La metaclassa <i>Artifact</i> es la especificación de una pieza física de información que se usa o produce en el desarrollo de software, y se define como una especialización de <i>Classifier</i> , por lo que se ha elegido como elemento del que extender estos estereotipos.
	Indicator	
Actor	Stakeholder	La metaclassa <i>Actor</i> de UML especifica cualquier ente (humano o no) que interactúa con el sistema, por lo que representa fielmente lo que pretendemos con <i>Stakeholder</i> .
ControlFlow	ProcessSequence	UML modela el flujo de ejecución mediante <i>ActivityEdges</i> y, por tanto, sus especializaciones. Una de ellas es precisamente <i>ControlFlow</i> , que es la que se define para una secuencia de <i>Actions</i> , por lo tanto es la metaclassa de la que extienden los elementos de secuencia de CBGProcess.
Dependency	isResponsible	Se ha optado por la metaclassa <i>Dependency</i> para extender estos esterotipos, como una forma de establecer la relación semántica que hay entre ellos.
	participes	
	entry	
	exit	
	isMeasure	
	contains	

Tabla 7.4: Justificación de las metaclasses en el perfil UML de CBGProcess

Con la definición del perfil UML de CBGProcess, desarrollados los elementos de diagramas y toolbox de EA, es posible generar un perfil ejecutable dentro de EA que permite la edición de modelos de procesos conformes a CBGProcess, atendiendo a la sintaxis concreta expuesta en la figura 7.6.

Initial	Final	Activity (Individual)	Activity (Collaborative)	Conditional
StakeHolder	Indicator (Individual)	Indicator (Collaborative)	Organization	Process
Product (Individual)	Product (Collaborative)	Metric (Individual)	Metric (Collaborative)	ProcessSequence
isResponsible	participes	entry	exit	contains

Figura 7.6: Sintaxis concreta de CBGProcess en CBG-Tool

La figura 7.7 muestra una captura de pantalla del editor de procesos colaborativos que forma parte de CBG-Tool, donde se puede observar la sintaxis concreta del lenguaje en el editor de la propia herramienta.

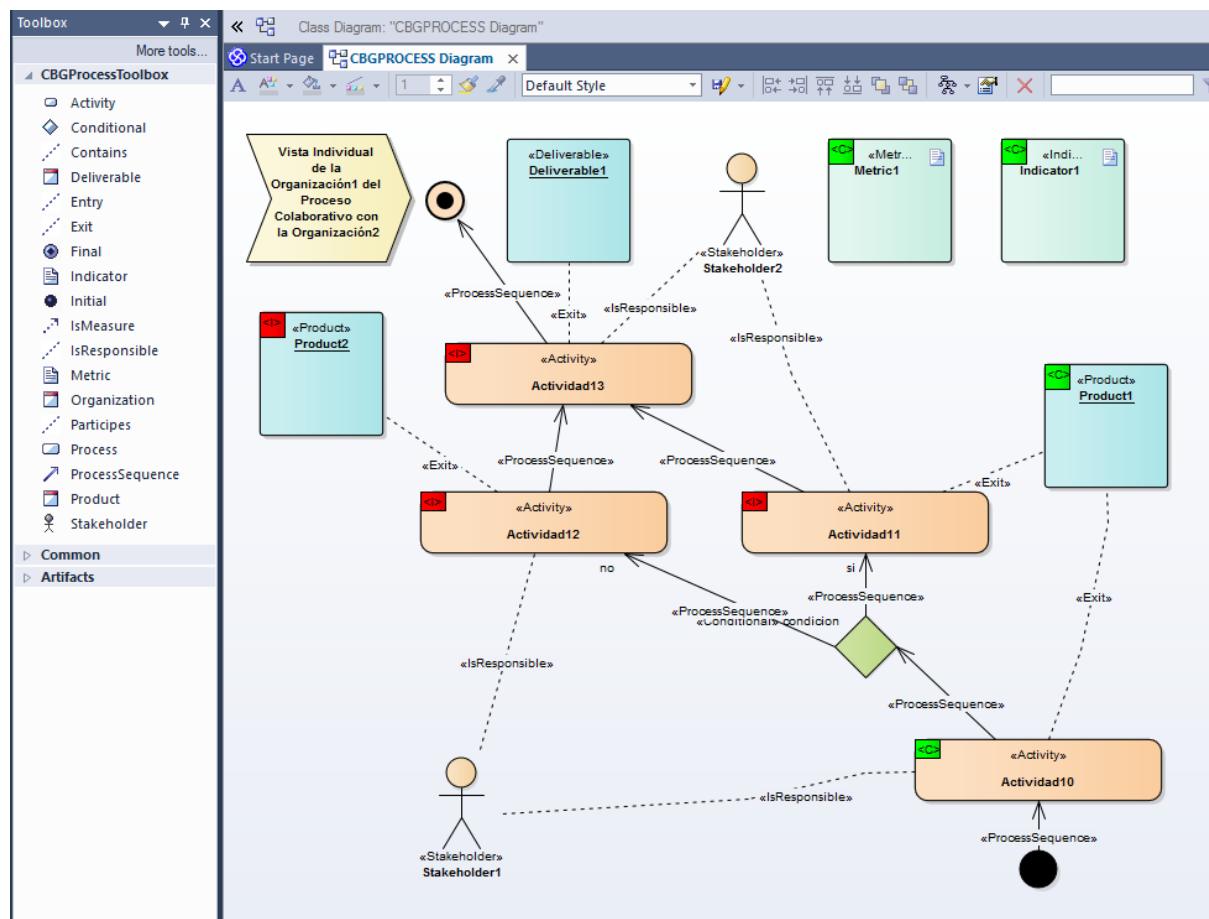


Figura 7.7: Editor de CBGProcess en CBG-Tool

7.2.2. El lenguaje de modelado de procesos colaborativos CBGCase en CBG-Tool

De igual forma que para el lenguaje CBGProcess, la incorporación del metamodelo CGB-CASE dentro de CBG-Tool se realiza a través de la definición de un perfil de UML, el cuál se observa en la figura 7.8.

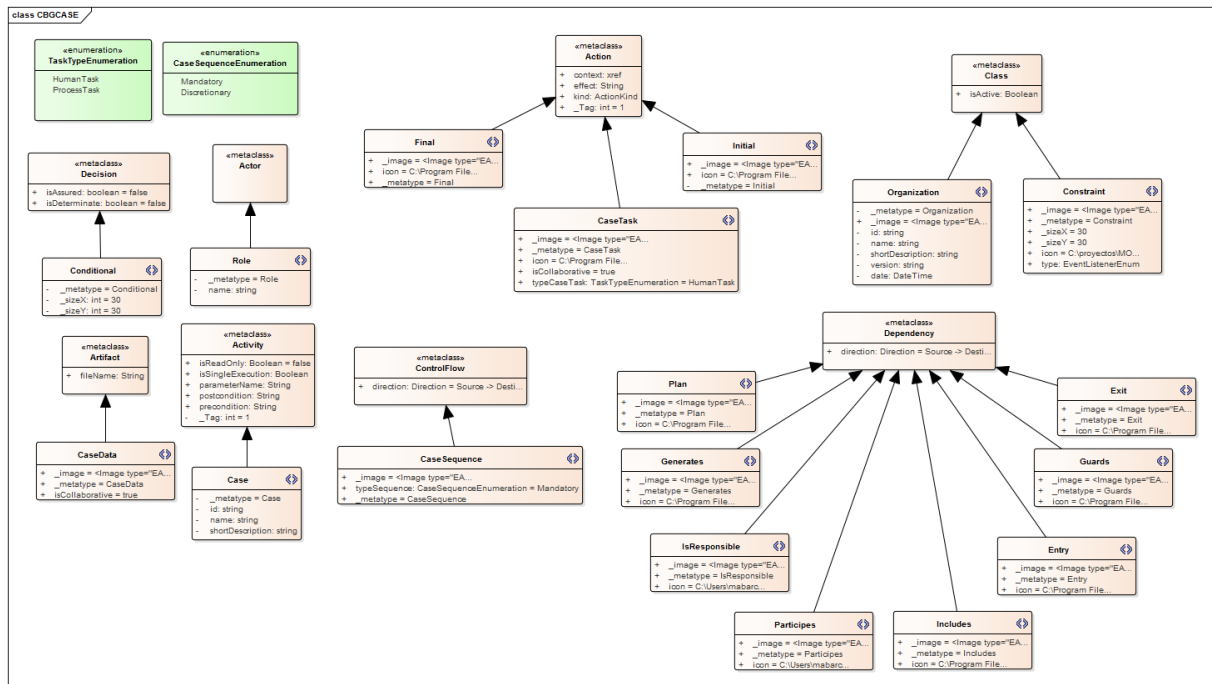


Figura 7.8: Perfil UML para CBGProcess

En la tabla 7.5 se muestran los estereotipos del perfil UML de CBGCase, así como la metaclass que se ha utilizado para realizarlo, y se justifica la elección de la metaclass correspondiente a cada estereotipo.

Metaclass UML	Estereotipo	Justificación
Activity	Case	La metaclass <i>Activity</i> representa la secuencia de tareas que formarán parte del caso, que podrán ser replanificadas en tiempo de ejecución atendiendo, y por ello ha sido elegida para ser extendida con <i>Case</i> .
Action	CaseTask	La metaclass <i>Action</i> representa un simple paso dentro de una <i>Activity</i> , lo que en CBGCase se ha denominado <i>CaseTask</i> .
	Initial	
	Final	
Decision	Conditional	Este elemento se corresponde a un <i>ControlNode</i> de UML, más concretamente los que hacen referencia a <i>DecisionNode</i> , que en EA está definido como <i>Decision</i> .
Class	Organization	La metaclass <i>Class</i> de EA coindidente con la Clase de UML se ha seleccionado para representar las <i>Organization</i> y <i>Constraint</i> de CBGCase.
	Constraint	
Artifact	CaseData	La metaclass <i>Artifact</i> es la especificación de una pieza física de información que se usa o produce en el proceso o caso, de software, y se define como una especialización de <i>Classifier</i> , por lo que se ha elegido como elemento del que extender estos estereotipos para el elemento <i>CaseData</i> de CBGCase.
Actor	Role	La metaclass <i>Actor</i> de UML especifica cualquier ente (humano o no) que interactúa con el sistema, por lo que representa fielmente lo que pretendemos con <i>Role</i> .
ControlFlow	CaseSequence	UML modela el flujo de ejecución mediante <i>ActivityEdges</i> y, por tanto, sus especializaciones. Una de ellas es precisamente <i>ControlFlow</i> , que es la que se define para una secuencia de <i>Actions</i> , por lo tanto es la metaclass de la que extienden los elementos de secuencia de CBGCase.
Dependency	isResponsible	Se ha optado por la metaclass <i>Dependency</i> para extender estos estereotipos, como una forma de establecer la relación semántica que hay entre ellos.
	plan	
	generates	
	isResponsible	
	participes	
	includes	
	entry	
	guards	
	exit	

Tabla 7.5: Justificación de las metaclasses en el perfil UML de CBGCase

Con la definición del perfil UML de CBGCase, desarrollados los elementos de diagramas y toolbox de EA, es posible generar un perfil ejecutable dentro de EA que permite la edición de modelos de procesos conformes a CBGCase, atendiendo a la sintaxis concreta expuesta en la figura 7.9.





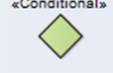


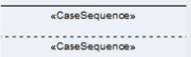

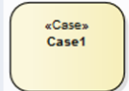


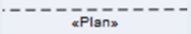
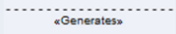
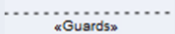
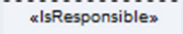
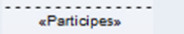
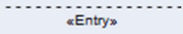
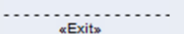
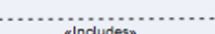
Initial	Final	CaseTask (Individual) (Human vs Process)	CaseTask (Collaborative) (Human vs Process)	Conditional
				
StakeHolder	Constraint	CaseSequence (Mandatory vs Discretionary)	Organization	Case
				
CaseData (Individual)	CaseData (Collaborative)	Plan	Generates	Guards
				
isResponsible	participes	entry	exit	includes
				

Figura 7.9: Sintaxis concreta de CBGCase en CBG-Tool

La figura 7.10 muestra una captura de pantalla del editor de casos colaborativos que forma parte de CBG-Tool, donde se puede observar la sintaxis concreta del lenguaje en el editor de la propia herramienta.

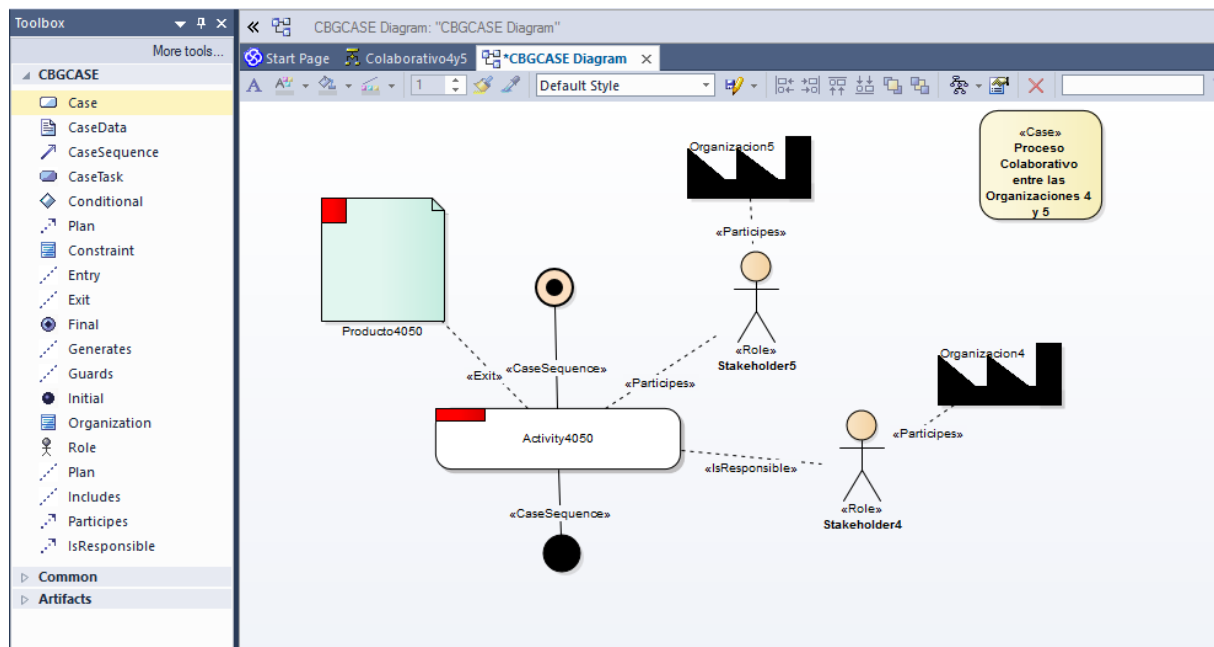


Figura 7.10: Editor de CBGCase en CBG-Tool

7.2.3. Implementación de las Transformación de INROMA al proceso colaborativo en CBG-Tool

En esta sección exponemos la forma en la que hemos implementado la transformación de modelos de procesos conformes a INROMA hacia CBGProcess. Para ello, hemos desarrollado en el lenguaje C# una librería que, desplegada como un *Add-In* dentro Enterprise Architect, implementa las transformaciones utilizando el menú contextual desde un diagrama de origen. A modo de ejemplo tomaremos la relación principal de la transformación expuesta en la sección 6.2.1 mostrada de nuevo en la figura 7.11.

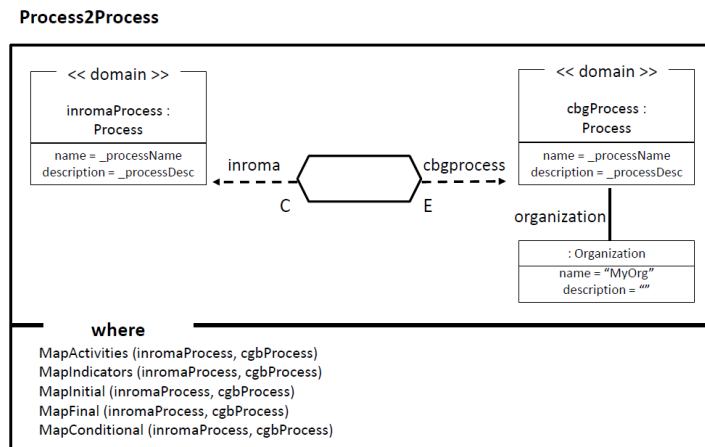


Figura 7.11: Vista QVT Relations de la relación Process2Process en la transformación M2M inroma2CBGProcess

Seguidamente exponemos esa misma relación atendiendo a la notación textual de QVT Relations, como se muestra en el algoritmo 7.1.

```

1  top relation Process2Process
2  {
3      _processName : String;
4      checkonly domain inromamodel inromaProcess : inroma::Process
5      {
6          name = _processName
7      };
8      enforce domain cbgmodel cgbProcess : cgb::Process
9      {
10         name = _processName
11     };
12     where {
13         MapActivities (inromaProcess , cgbProcess);
14         MapIndicators (inromaProcess , cgbProcess);
15         MapInitial (inromaProcess , cgbProcess);
16         MapFinal (inromaProcess , cgbProcess);
17         MapConditional (inromaProcess , cgbProcess);
18     }
19 }

```

Algoritmo 7.1: Relación principal en QVT textual para la transformación INROMA2CBGProcess

Así, una vez modelada una transformación con la sintaxis concreta de QVT Relations, será necesario incorporar con *Add-Ins* el código fuente equivalente, tal y como se expone en el algoritmo 7.2.

```

1  void Process2Process(EA.Element inromaProcess, EA.Element cbgProcess)
2  {
3      if (cbgProcess == null)
4          cbgProcess = addNewProcess(inromaProcess.Name);
5
6      MapActivities(inromaProcess, cbgProcess);
7      MapIndicators(inromaProcess, cbgProcess);
8      MapInitial(inromaProcess, cbgProcess);
9      MapFinal(inromaProcess, cbgProcess);
10     MapConditional(inromaProcess, cbgProcess);
11     MapOrganization(cbgProcess);
12 }

```

Algoritmo 7.2: Implementación de la transformación principal INROMA2CBGPROCESS en C#

El anexo A expone el código fuente que implementa todas las relaciones de la transformación entre M2M entre INROMA y CBGProcess dentro de la solución CBG-Tool.

7.2.4. Implementación de las Transformación del proceso al caso colaborativo en CBG-Tool

En esta sección exponemos la forma en la que hemos implementado la transformación de modelos de procesos conformes a CBRProcess hacia CBGCase. Para ello, hemos desarrollado en el lenguaje C# una librería que, desplegada como un *Add-In* dentro Enterprise Architect, implementa las transformaciones utilizando el menú contextual desde un diagrama de origen. A modo de ejemplo tomaremos la relación principal de la transformación expuesta en la sección 6.2.2 mostrada de nuevo en la figura 7.12.

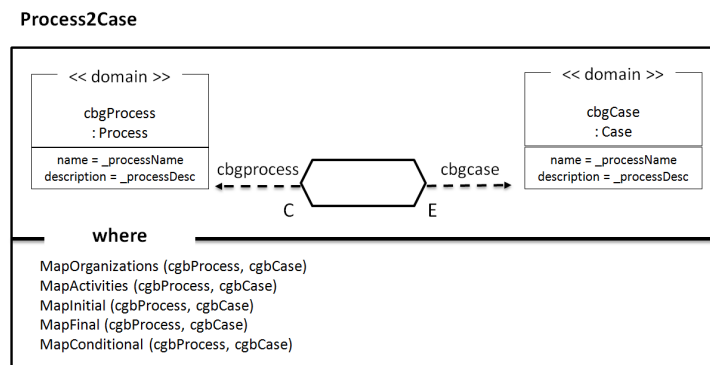


Figura 7.12: Vista QVT Relations de la relación Process2Case en la transformación M2M CBG-Process2CBGCase

Seguidamente exponemos esa misma relación atendiendo a la notación textual de QVT Relations, como se muestra en el algoritmo 7.3.

```

1  top relation Process2Case
2  {
3      _processName : String;
4      checkonly domain cbgProcessmodel cbgProcess : cbg::Process
5      {
6          name = _processName
7      };
8      enforce domain cbgCasemodel cbgCase : cbg::Case
9      {
10         name = _processName
11     };
12     where {
13         MapOrganizations (cbgProcessmodel , cbgCasemodel);
14         MapActivities (cbgProcessmodel , cbgCasemodel);
15         MapInitial (cbgProcessmodel , cbgCasemodel);
16         MapFinal (cbgProcessmodel , cbgCasemodel);
17         MapConditional (cbgProcessmodel , cbgCasemodel);
18     }
19 }

```

Algoritmo 7.3: Relación principal en QVT textual para la transformación CBGProcess2CBGCase

Así, una vez modelada una transformación con la sintaxis concreta de QVT Relations, será necesario incorporar con *Add-Ins* el código fuente equivalente, tal y como se expone en el algoritmo 7.4.

```

1  void Process2Case(EA.Element cbgProcess , EA.Element cbgCase)
2  {
3      if (cbgCase == null)
4          cbgCase = addNewCase(cbgProcess.Name);
5
6      MapOrganizations(cbgProcess , cbgCase);
7      MapActivities(cbgProcess , cbgCase);
8      MapInitial(cbgProcess , cbgCase);
9      MapFinal(cbgProcess , cbgCase);
10     MapConditional(cbgProcess , cbgCase);
11 }

```

Algoritmo 7.4: Implementación de la transformación principal CBGProcess2CBGCase en C#

El anexo B expone el código fuente que implementa todas las relaciones de la transformación entre M2M entre CBGProcess y CBGCase dentro de la solución CBG-Tool.

7.3. Validación de CBG-Tool

Para validar la herramienta vamos a hacer uso de los escenarios que fueron descritos como muestra de la necesidad en la sección 3.2 y que resumimos a continuación:

1. La creación de modelos de procesos en colaboración reutilizando modelos de procesos individuales. A partir de la descripción de los procesos de cada organización, establecer el modelo colaborativo y permitir que cada entidad disponga de una vista detallada en la que se detallen los elementos individuales y los del modelo en colaboración.
2. La inclusión de nuevas organizaciones sobre una CdS ya formada que tiene su proceso colaborativo modelado, como forma de validar la flexibilidad de la solución para incorporar el dinamismo natural de las CdS, incorporando nuevos procesos a la colaboración sin por ello tener que empezar de cero o condicionar el proceso colaborativo ya disponible.
3. La flexibilidad de un proceso individual para recoger la participación de la organización en diversas CdS en las que puede cumplir un rol diferente, facilitando la integración entre la gestión de procesos individual y la global con un enfoque de CdS.
4. Adicionalmente validaremos la solución desde el dominio de casos colaborativos, para poder dar soporte a la definición de la secuencia de acciones en tiempo de ejecución.

Seguidamente pasamos a detallar la validación de cada uno de los escenarios con la herramienta CBG-Tool.

7.3.1. Crear la vista colaborativa a partir de un modelo de proceso individual

En primer lugar vamos a presentar el escenario en el que dos organizaciones crean un modelo de proceso colaborativo a partir de sus modelos de procesos individuales. Como hemos comentado en la sección 7.1.1 partiremos de los modelos de procesos en INROMA, sin detallar el uso de la herramienta MONETA. Las figuras 7.13 y 7.14 exponen un modelo de proceso individual de las organizaciones 1 y 2 respectivamente.

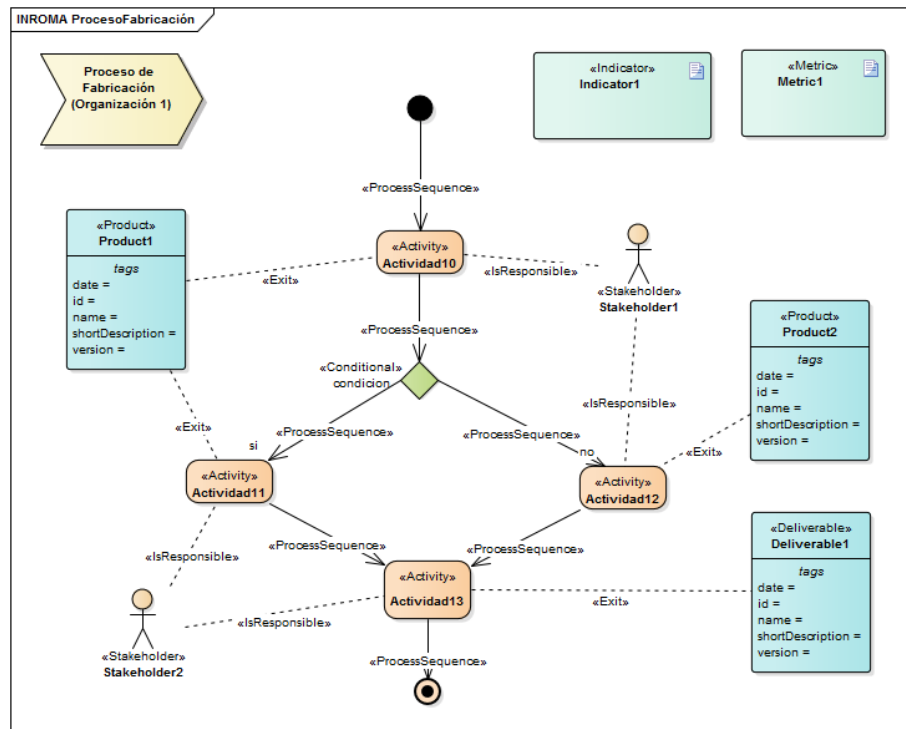


Figura 7.13: Modelo de proceso individual de la Organización 1

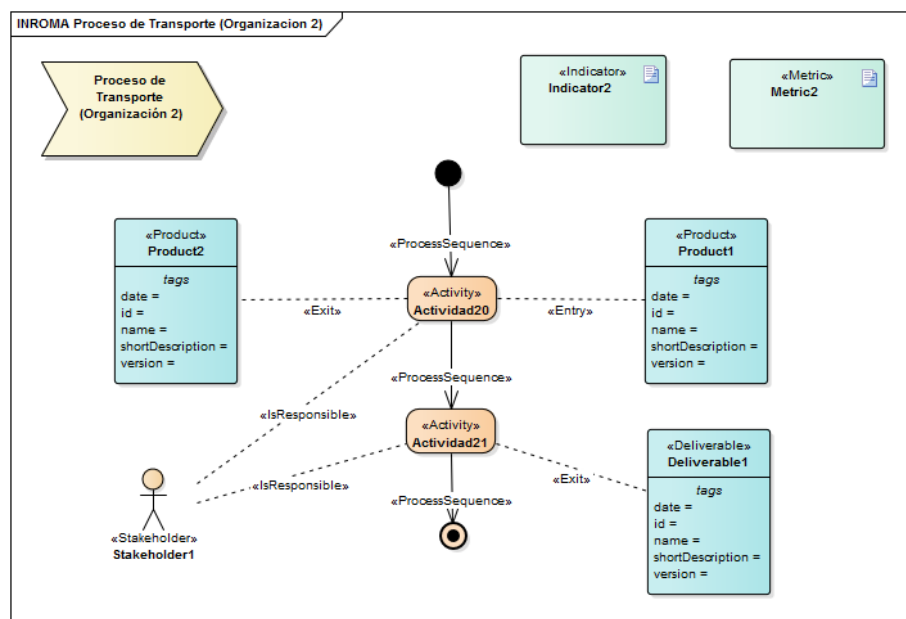


Figura 7.14: Modelo de proceso individual de la Organización 2

Con la ayuda de CBG-Tool hacemos uso de las transformaciones para generar la vista que cada una de las organizaciones tiene de la colaboración. De esta forma, la figura 7.15 muestra la vista individual de la organización 1 en el proceso colaborativo. Respecto al modelo anterior presentado en la figura 7.13.

En este caso la *Organización1* establece que la *Activity10* formará parte de la colaboración, así como el *Product1* resultado de la misma. De igual forma el *Indicator1* y la *Metric1* quedan establecidas como elementos dentro de la colaboración. El resto de elementos del proceso quedan en la vista individual y no serán visibles en el modelo de proceso colaborativo.

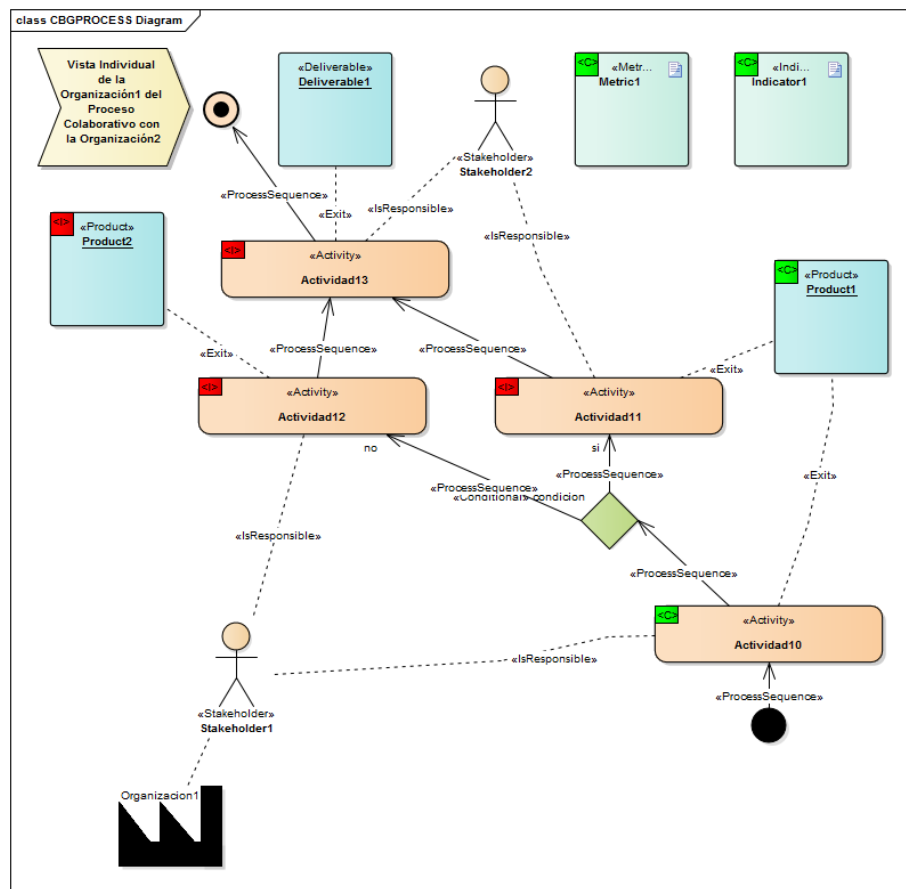


Figura 7.15: Vista individual de la Organización 1 sobre el proceso en colaboración

De igual forma la *Organización2* incorpora la *Activity10* colaborativa, con su *Product1* resultado, si bien en este caso no coincide exactamente con la actividad y producto a desarrollar de forma individual. Por ello, hace uso del mecanismo de CBGProcess para seguir manteniendo la *Activity20* de forma privada pero establecer la trazabilidad con la colaboración a través de la asociación *Contains*. La figura 7.16 muestra la vista individual de la *Organización2* sobre el proceso en colaboración.

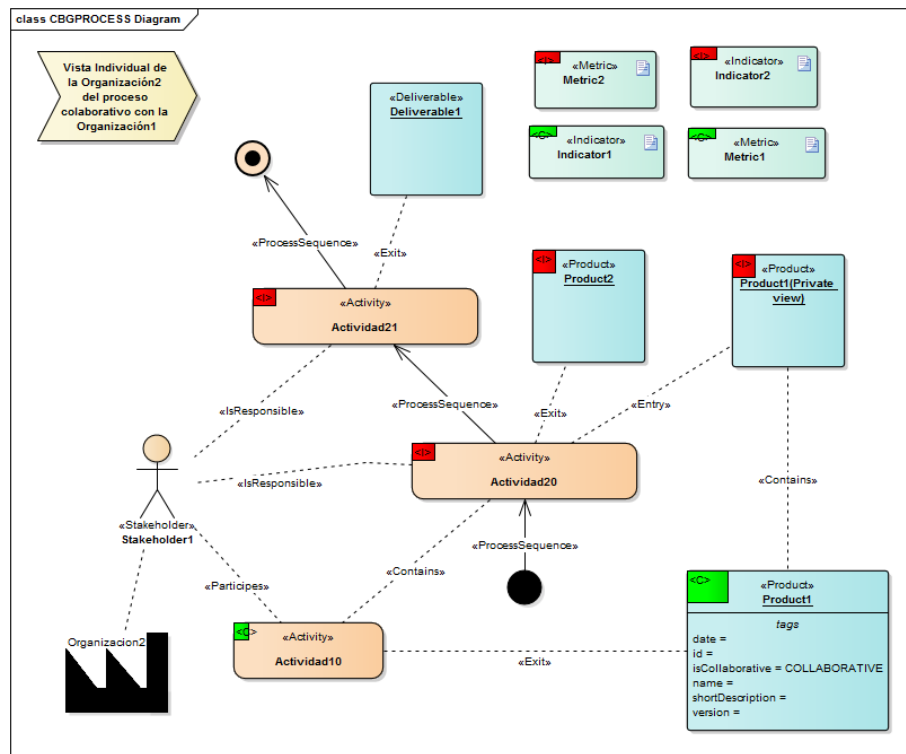


Figura 7.16: Vista individual de la Organización 2 sobre el proceso en colaboración

Por último, se muestra el resultado del proceso colaborativo en la figura 7.17.

De esta forma validamos la creación de modelos de procesos en colaboración reutilizando modelos de procesos individuales.

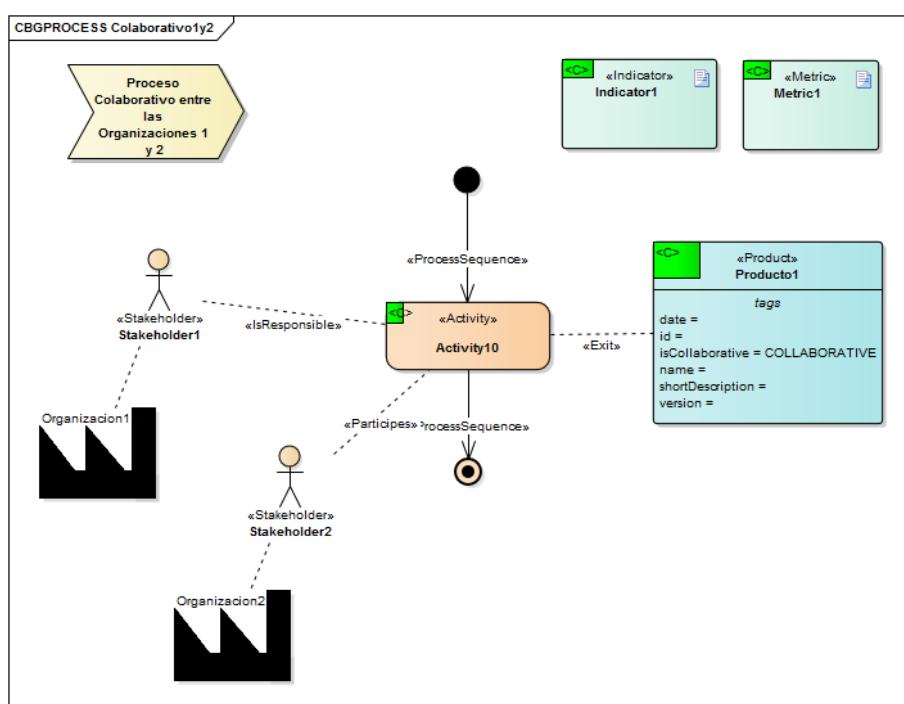


Figura 7.17: Modelo de proceso colaborativo entre las Organizaciones 1 y 2

7.3.2. Incorporar nuevos procesos al modelo de proceso colaborativo

En este escenario tenemos una nueva *Organizacion3* que quiere formar parte de la colaboración definida entre la *Organizacion1* y la *Organizacion2* mostrada en la figura 7.17.

Como en el caso anterior, la *Organizacion3* tiene su propio proceso individual modelado en INROMA y representado en la figura 7.18.

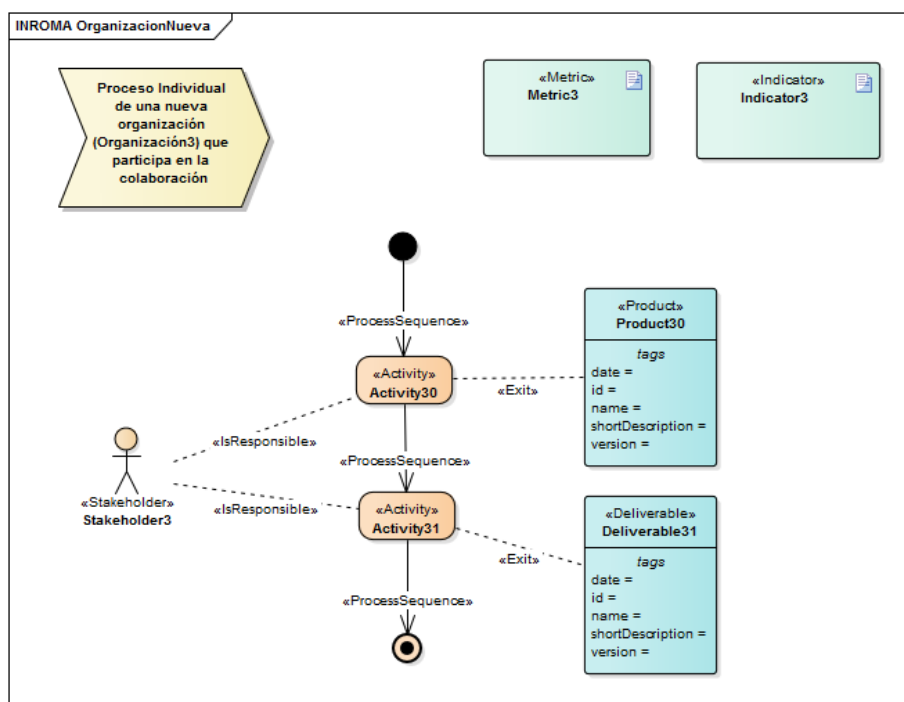


Figura 7.18: Modelo de proceso individual de una nueva Organización 3

Atendiendo a la colaboración deseada, la *Organizacion3* genera su vista individual de la colaboración, en la que la *Activity10* queda incluida pero reflejada con el nombre *Activity1030*, ya que al ser la vista individual puede ser descrita de la forma en la que esa organización concreta desee. Para representar un escenario diferente al anterior, suponemos que el resultado de esa actividad tiene una componente colaborativa representada por el *Product1030*, pero a nivel individual se mantendrá el *Product30* de forma privada, si bien la relación *Contains* le permitirá tener la trazabilidad de que es un elemento que de alguna forma contribuye en la colaboración.

Esta vista individual de la *Organizacion3* sobre el proceso en colaboración se expone en la figura 7.19. La figura 7.20 muestra el nuevo proceso colaborativo en el que la *Organizacion3* aparece como nuevo *Stakeholder* en la *Activity10*.

De esta forma validamos la inclusión de nuevas organizaciones sobre una CdS ya formada que tiene su proceso colaborativo modelado, incorporando nuevos procesos a la colaboración sin por ello tener que empezar de cero o condicionar el proceso colaborativo ya disponible.

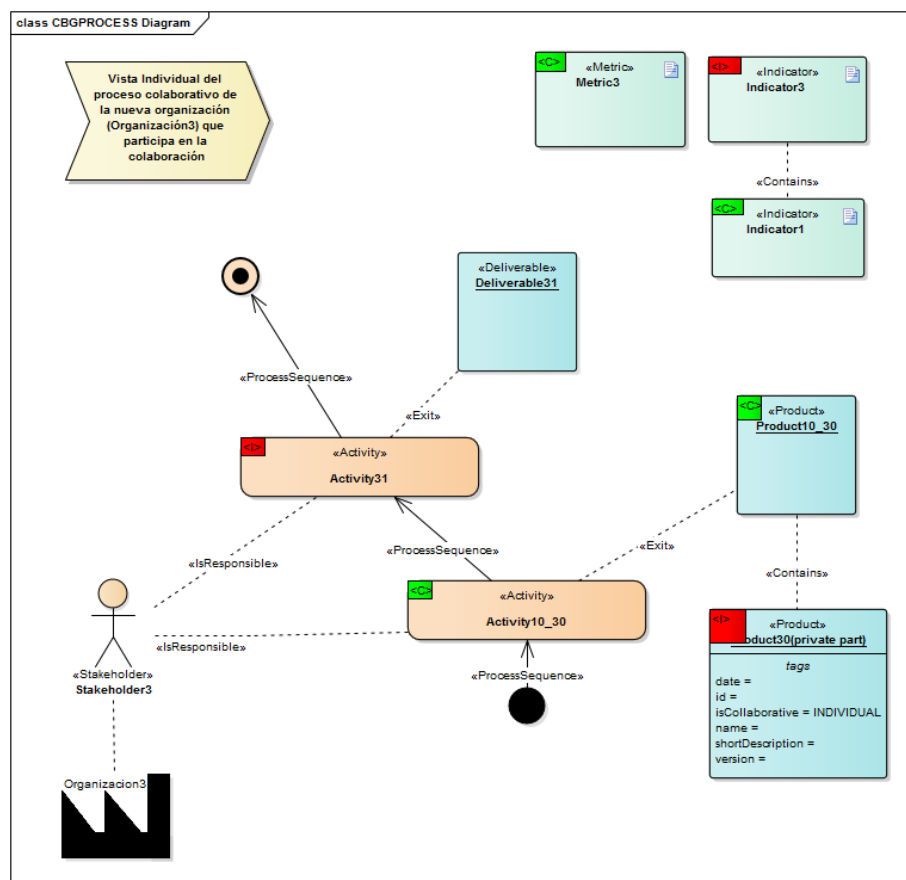


Figura 7.19: Vista individual de la Organización 3 sobre el proceso en colaboración

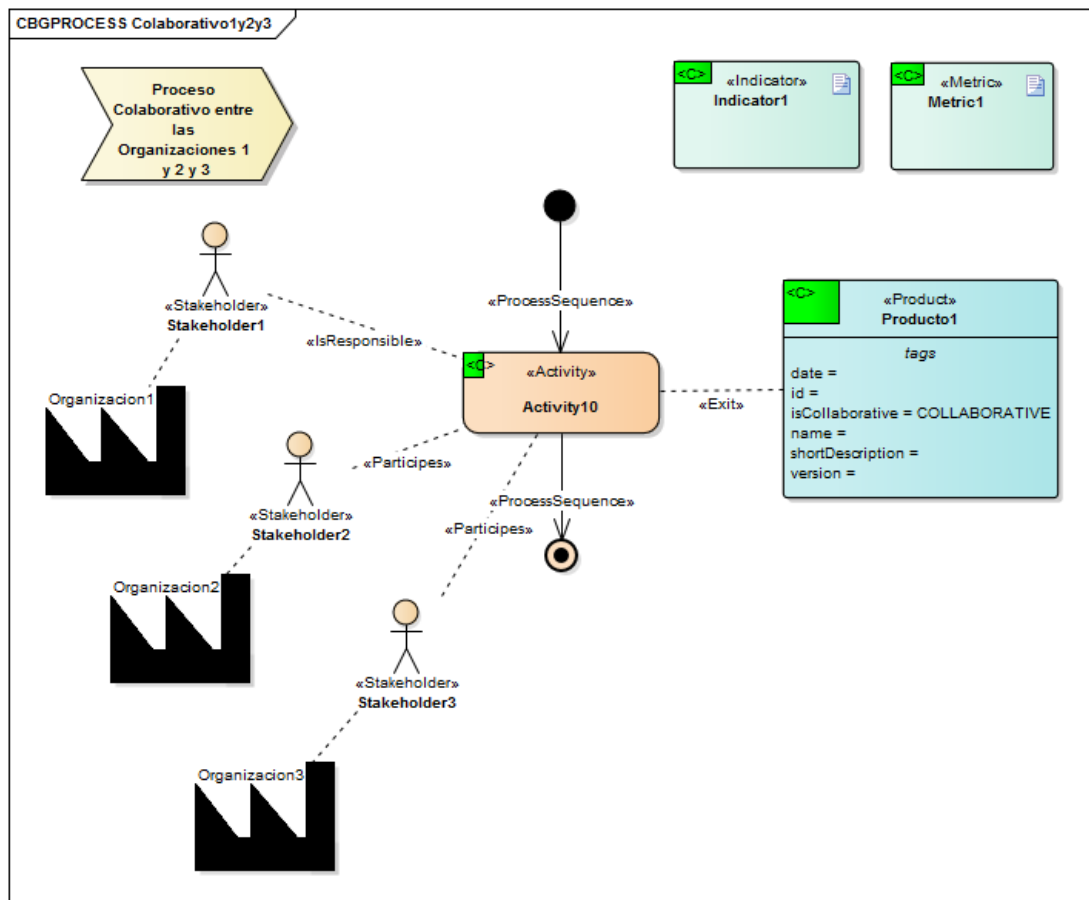


Figura 7.20: Modelo de proceso colaborativo entre las Organizaciones 1, 2 y 3

7.3.3. Un modelo de proceso individual en varios procesos colaborativos

En este escenario supongamos que tenemos una CdS colaborativa entre la *Organizacion4* y la *Organizacion5* representado por la figura 7.21.

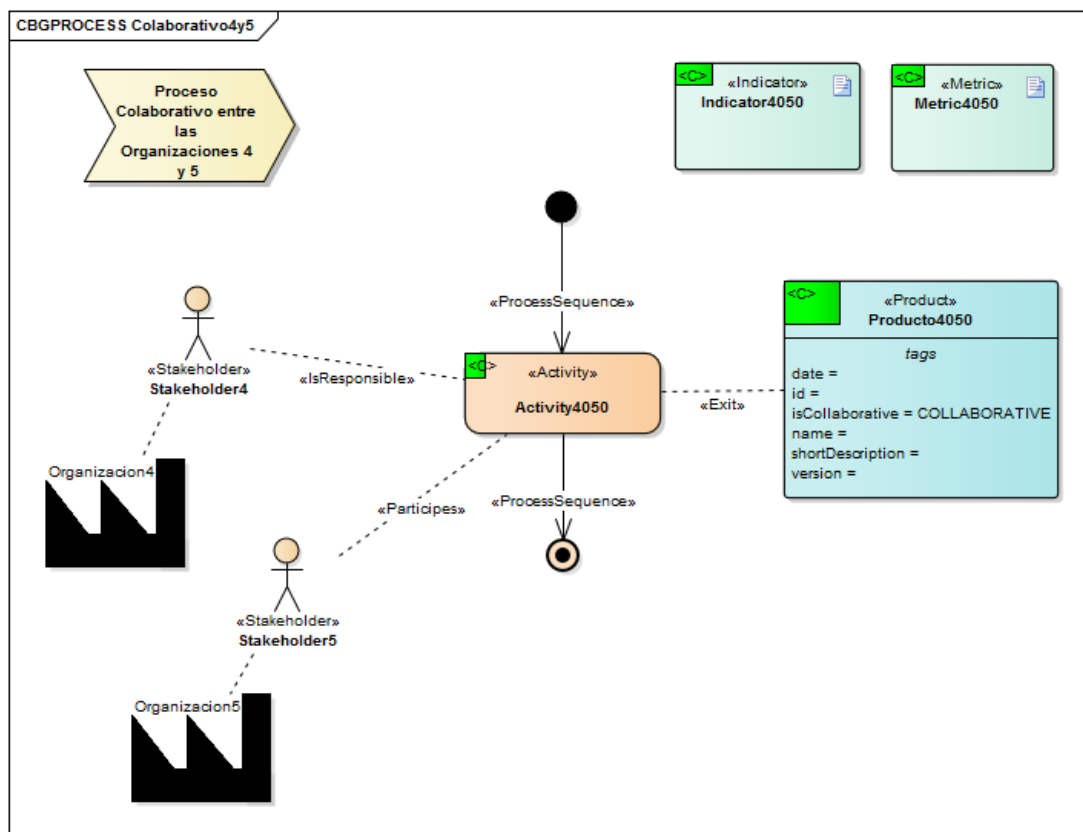


Figura 7.21: Modelo de proceso colaborativo entre las Organizaciones 4 y 5

La *Organizacion1* que presentamos anteriormente y que tiene una vista del proceso en colaboración con la *Organizacion2*, tal y como detallamos en la figura 7.17, desea participar en esta nueva CdS. Para ello, debe modificar su vista para incluir los cambios necesarios que le permitan incluir los nuevos elementos fruto de la colaboración con las organizaciones 4 y 5.

La figura 7.22 muestra esos cambios, en los que podemos observar que aparecen los elementos comunes de la nueva colaboración (*Activity4050* y *Product4050*) y se establecen las relaciones con las actividades y productos que se mantienen de forma privada (*Actividad12* y *Product2* respectivamente).

De esta forma validamos la flexibilidad de un proceso individual para recoger la participación de la organización en diversas CdS en las que puede cumplir un rol diferente, facilitando la integración entre la gestión de procesos individual y la global con un enfoque de CdS.

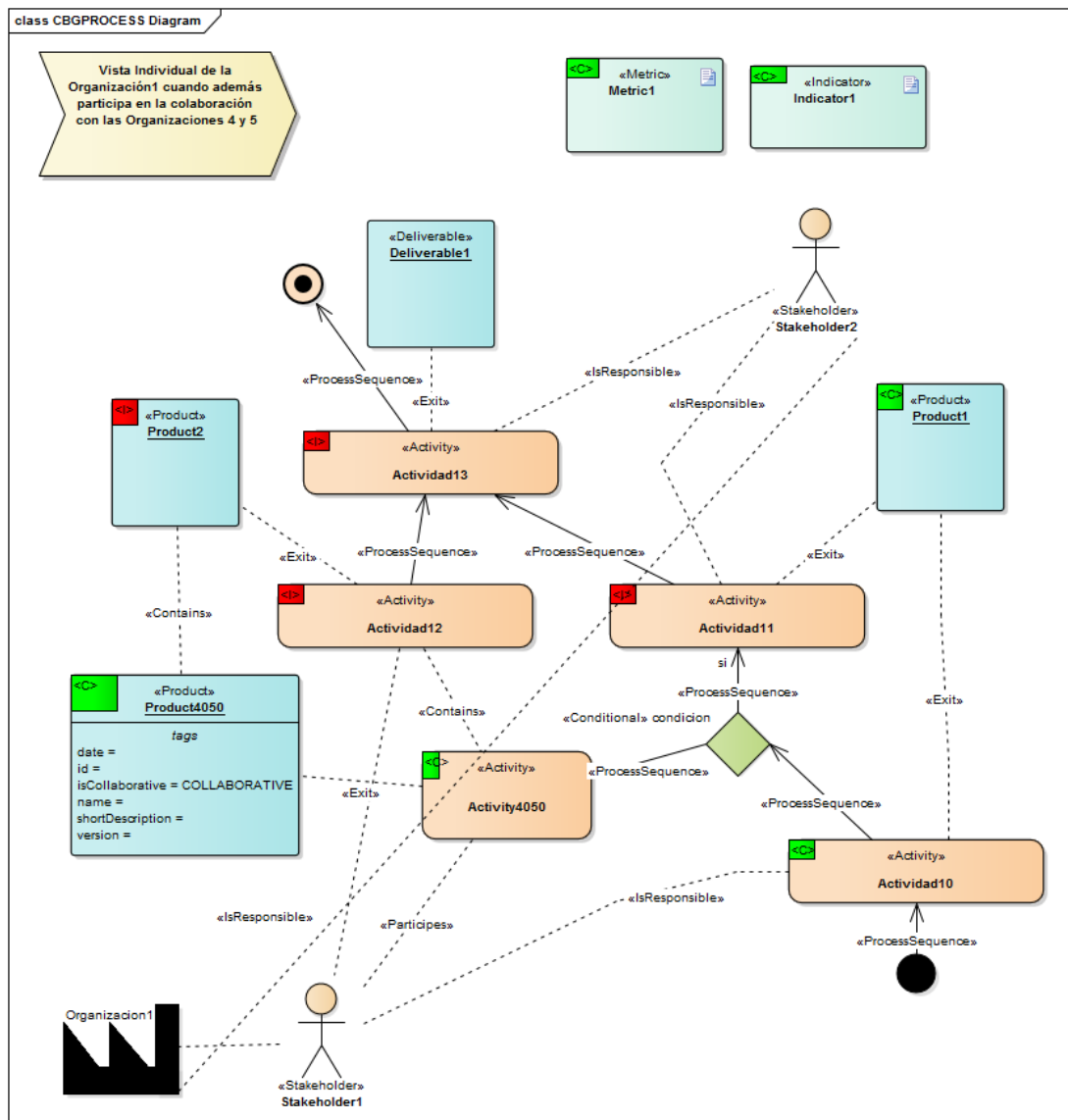


Figura 7.22: Vista individual de la Organización 1 sobre el proceso en colaboración, tanto en la cadena 1 y 2 como con las organizaciones 4 y 5

7.3.4. Crear el modelo del caso colaborativo

El último escenario de validación corresponde al dominio de casos colaborativos, para poder dar soporte a la definición de la secuencia de acciones en tiempo de ejecución.

Para ello, supongamos que en el escenario del proceso colaborativo de las Organizaciones 1, 2 y 3 descrito en la figura 7.20 deseamos añadir la capacidad de planificación dinámica de las tareas. En este caso, haremos uso de la transformación CBGProcess2CBGCase implementada en la solución CBG-Tool y obtendremos de forma automática el modelo de caso colaborativo equivalente, expuesto en la figura 7.23.

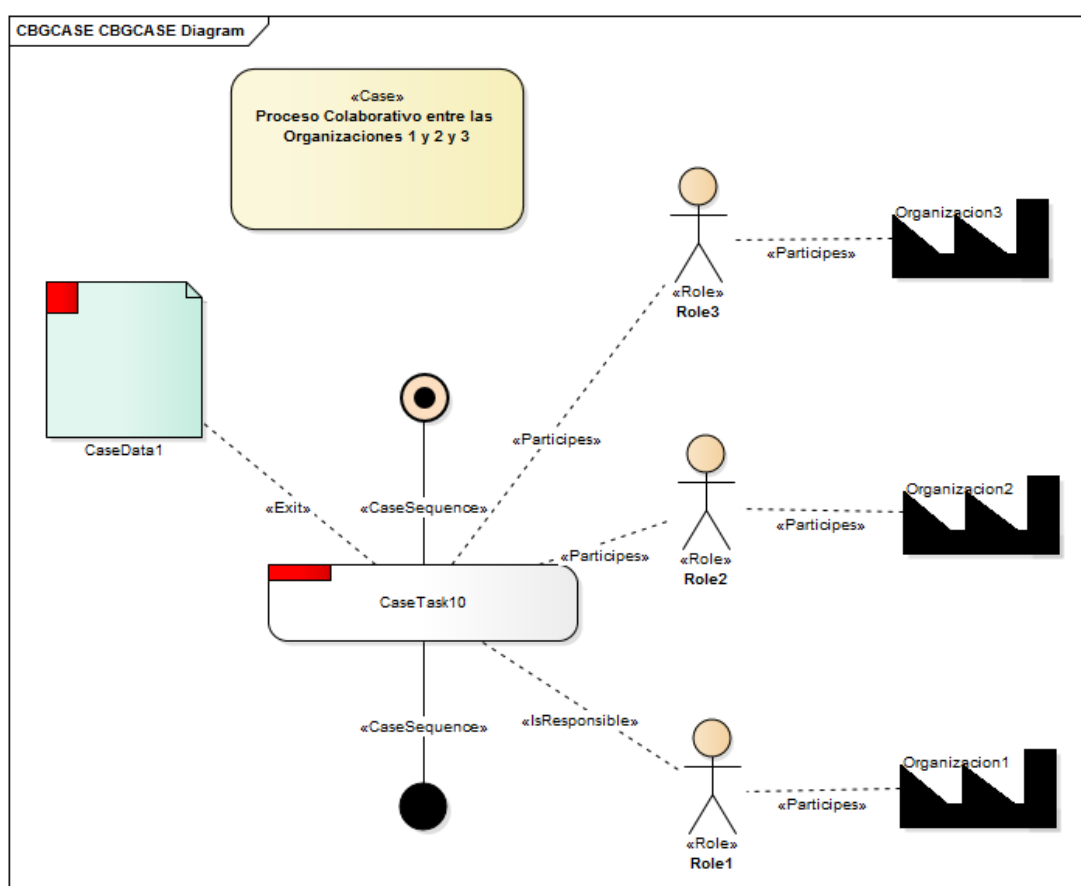


Figura 7.23: Modelo de caso colaborativo entre las Organizaciones 1, 2 y 3

A partir del modelo de caso colaborativo, podremos generar la visión individual de cada organización dentro de la colaboración, bien a partir del modelo del caso o a partir del modelo de vista individual del proceso colaborativo, ya que en ambos casos los metamodelos y la solución del marco CBG mantienen la trazabilidad de los elementos. A modo de ejemplo, a partir de la vista individual de la organización 1 en la colaboración, mostrada en la figura 7.15, a través de CBG-Tool somos capaces de generar de forma automática su vista respecto al caso colaborativo, tal y como expone la figura 7.24.

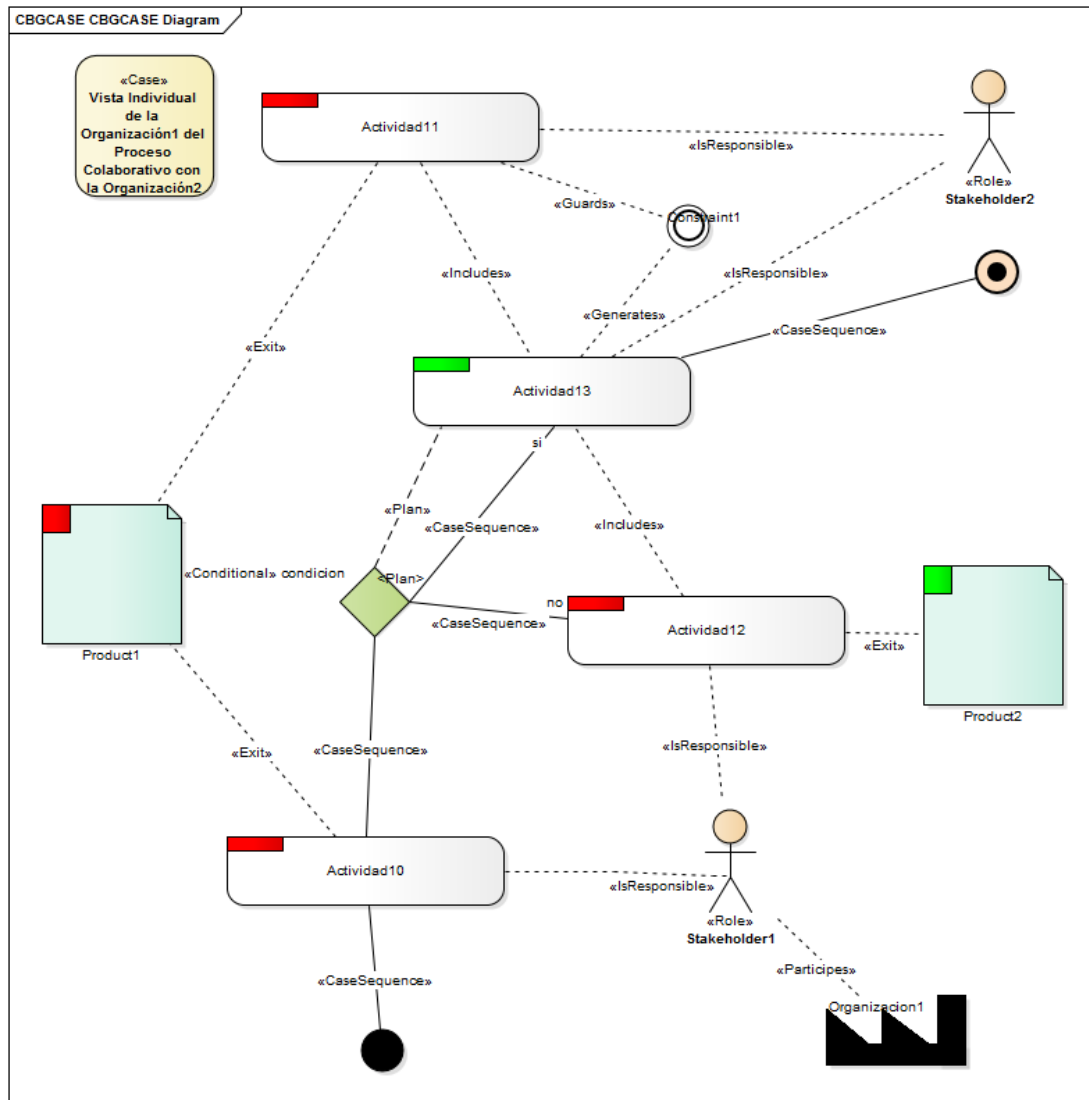


Figura 7.24: Vista individual de la Organización 1 sobre el caso en colaboración entras las organizaciones 1, 2 y 3

De esta forma hemos podido comprobar cómo la solución CBG-Tool da soporte para pasar del dominio de procesos al de casos, añadiendo las capacidades de planificación dinámica requeridas en ciertos casos, manteniendo la trazabilidad tanto con los modelos de procesos colaborativos como con la vista individual que cada organización pudiera tener respecto al proceso o caso colaborativo.

7.4. Conclusiones

En este capítulo se ha presentado CBG-Tool como la herramienta de soporte del marco CBG que hemos ido definiendo y desarrollando a lo largo de este trabajo de tesis. CBG-Tool se configura como la solución para facilitar la creación de modelos y la ejecución de las transformaciones desde un punto de vista práctico.

Hemos detallado sus funcionalidades a partir de la descripción de sus casos de uso, presentado la arquitectura de la solución dentro del contexto de ejecución de Enterprise Architect. Seguidamente hemos expuesto cómo CBG-Tool da soporte a la edición de modelos conformes a CBGProcess y CBGCase, a partir de la definición de sus perfiles UML y la creación de su sintaxis concreta. A continuación hemos concretado la forma en la que se ha desarrollado un Add-in que implementa las transformaciones dentro de la solución.

Por último hemos querido validar las funcionalidades de CBG-Tool atendiendo a escenarios ficticios que representan las situaciones reales que han justificado la realización de este trabajo de tesis y que fueron descritas en los capítulos iniciales, haciendo uso paso a paso de la solución descrita en el capítulo.

Una vez hemos presentado y validado la solución CBG-Tool con estos casos ficticios, será en el siguiente capítulo dónde se expondrán los casos de estudio reales que nos han permitido validar la el marco CBG como solución para el modelado de procesos y casos colaborativos.

Capítulo 8

Evaluación del marco CBG en entornos reales: caso de estudio

Una vez hemos determinado el problema a resolver y se han ido definiendo los diversos elementos que forman parte del marco CBG propuesto para crear procesos y casos de negocio colaborativos, es el momento de evaluar la solución propuesta en casos de estudio reales.

Este capítulo tiene por objetivo evaluar, mediante un caso de estudio concreto extraído de un proyecto real con la industria, que tanto la definición del problema planteado tiene lugar como que los resultados y las contribuciones resultado de este trabajo de tesis habrían facilitado el desarrollo de los mismos. Cabe destacar que fue uno de estos proyectos lo que evidenció la necesidad contra la que trabajamos y, en los trabajos futuros, vamos a poder aplicar los resultados aquí obtenidos facilitando en gran medida la ejecución de los mismos.

El capítulo se estructura como sigue. La primera sección establece el contexto general del proyecto real seleccionado. A continuación, se expone en sí el caso de estudio, atendiendo al uso del marco CBG. Para finalizar, en la última sección se resumen las principales conclusiones obtenidas a lo largo del capítulo.

8.1. Elección del caso de estudio

A lo largo de los capítulos anteriores hemos puesto en contexto la necesidad que pretende cubrir esta tesis, de forma resumida:

- Las organizaciones definen sus procesos de negocio, de forma individual, pero luego en el mundo real interactúan con otras organizaciones, clientes y proveedores, formando cadenas de valor empresarial comúnmente conocidas como CdS.
- Para definir sus procesos de negocio, las organizaciones eligen de entre la gran cantidad de lenguajes de modelado existentes, apostando por un método para rediseñar, monitorizar o incluso ejecutar sus procesos de negocio individuales.
- Para poder tener una vista de los procesos de negocio a lo largo de toda la CdS, lo habitual es generar desde cero un nuevo modelo de proceso de negocio que incorpora las actuaciones llevadas a cabo por los diversos participantes. Este enfoque obliga a duplicar el trabajo y a mantener la consistencia entre esa visión individual de su actividad y la integrada en cada CdS en la que participe.
- Las CdS son muy dinámicas, entran y salen participantes, hay nuevos comportamientos y políticas, las decisiones se toman de forma distribuida, por lo que es preciso disponer de modelos de procesos flexibles y adaptables, que evolucionen garantizando esa consistencia con la visión individual.

En base a esta problemática ello hemos desarrollado una solución que nos permita crear modelos de procesos de negocio de la CdS a partir de la integración y de las interacciones existentes entre los procesos de negocio individuales de los diversos participantes. Para crear estos modelos, hemos propuesto un nuevo enfoque, bottom-up, en el que se dé soporte a una diversidad de lenguajes y herramientas y se reutilicen los modelos existentes.

Como resultado el marco CBG aborda el problema de modelar los procesos de negocio de la CdS a partir de los modelos de procesos de negocio individuales que una organización pueda tener, con cualquier lenguaje de modelado de procesos, de forma que añadamos las capacidades de flexibilidad y adaptación de los modelos de procesos necesarias en el dominio de la CdS.

Entre los objetivos planteados en la sección 3.3 estaba el validar la propuesta con casos de estudio reales. Este trabajo de tesis se ha llevado a cabo dentro del grupo de investigación Ingeniería Web y Testing Temprano (IWT2) de la Universidad de Sevilla. IWT2 colabora de forma habitual con entidades, tanto públicas como privadas, en dominios muy diversos mediante diferentes proyectos de I+D, por lo que el grupo ha facilitado en gran medida la consecución de dicho objetivo, ya que nos ha proporcionado casos y situaciones en empresas en los que la problemática aquí enunciada era una realidad. De igual forma, esta estrecha relación con al industria garantiza la posibilidad de transferencia inmediata de los resultados de la tesis a nivel práctico.

De entre los diversos casos de estudio potenciales, hemos seleccionado aquellos en los que pudiéramos evidenciar no sólo la necesidad de fondo sino los diversos escenarios planteados. Por ello nos hemos basado en reproducir la problemática acontecida dentro de *ITChain*: proyecto para el desarrollo de soluciones de tecnologías de la información para la CdS Colaborativa. Este proyecto se ha llevado a cabo en el marco del PLAN AVANZA 2 con la cofinanciación del Ministerio de Industria, Energía y Turismo y la colaboración de seis entidades. La tabla 8.1 resume la información más relevante relativa a dicho proyecto.

ITChain: Tecnologías de la información para la cadena de suministro Colaborativa	
Investigador Principal:	Miguel Ángel Barcelona Liédana
Referencia:	TSI-020302-2010-80
Fecha de inicio:	01/09/2010
Fecha de fin:	31/03/2013
Empresa / Organismo financiador:	Ministerio de Industria, Energía y Turismo a través del PLAN AVANZA
Presupuesto:	3.803.868,75€

Tabla 8.1: Información detallada del proyecto ITChain

Una vez hemos puesto en contexto el proyecto seleccionado para la validación del marco CBG, en las siguientes secciones detallaremos el método de evaluación, exponiendo su objetivo, la necesidad detectada en relación al problema planteado en este trabajo de tesis, cómo fueron abordados en su momento y, finalmente, cómo lo habrían sido si hubieran contado con el marco de referencia CGB resultado de este trabajo.

8.2. Caso de estudio: ITCHAIN

El proyecto ITChain persigue la creación de una infraestructura para catalogar, publicar, descubrir, ejecutar y componer dinámicamente servicios en movilidad y dependientes del contexto, que facilite la interoperatividad de los sistemas existentes en una cadena de suministro y dote a los diversos agentes que la componen de información global para tomar decisiones de forma colaborativa. Esta plataforma multisectorial, tiene especial aplicación en los escenarios en los que existan varios participantes y además su número y comportamiento cambie de forma dinámica, como ocurre con carácter general en las cadenas de suministro integrales.

Para el desarrollo del proyecto contó con la colaboración de las siguientes entidades:

1. ATOS SPAIN S.A. (ATOS)
2. CENTROS COMERCIALES CARREFOUR S.A. (CARREFOUR)
3. INGENIERÍA DE SISTEMAS PARA LA DEFENSA DE ESPAÑA S.A. (ISDEFE)
4. MOVILITAS BUSINESS COACHING S.L. (MOVILITAS)
5. CAR VOLUM S.L. (CARVOLUM)
6. INSTITUTO TECNOLÓGICO DE ARAGÓN (ITA)

A nivel sectorial, el proyecto tenía los siguientes objetivos:

- **Modelar los procesos logísticos de las organizaciones y simular los escenarios**, para definir qué agentes forman la CdS, qué información es relevante para cada uno, quién la genera, dónde y cómo se almacena, quién, cuándo y cómo se actualiza.
- Utilizar la infraestructura como una herramienta para mejorar la toma de decisiones, al disponer de una mayor visibilidad a lo largo de la CdS.
- Establecer, mediante la ejecución de los procesos en la infraestructura, un cuadro de métricas e indicadores que permita valorar de forma objetiva la ganancia obtenida en un sector concreto, como el de la gran distribución de productos alimentarios, el tomar las decisiones con mayor volumen de información actualizada.
- Favorecer una logística colaborativa, en la que el intercambio de información permita optimizar las operaciones a nivel global, haciendo que los diversos participantes de la CdS sean más eficientes y por tanto más competitivos.
- Permitir una operativa logística más sostenible, que optimice las rutas de distribución y transporte, reduciendo las emisiones contaminantes y mejore el medio ambiente.

Una vez hemos visto el contexto general del proyecto, nos vamos a centrar en el objetivo específico de cómo modelar los procesos de las organizaciones, de cara a realizar una simulación de escenarios atendiendo a la colaboración en la CdS, para lo cual es preciso obtener el modelo de proceso colaborativo. En particular, vamos a centrarnos en uno de los casos piloto del proyecto, elaborado conjuntamente entre CARREFOUR e ITA.

El caso concreto cubre el modelado del proceso colaborativo de compra y distribución de pescado fresco, desde las lonjas hasta el punto de venta, pasando por unas plataformas logísticas de manipulación y almacenaje del producto dispersas por todo el territorio nacional. En este escenario, participan el comprador en la lonja, que recibe un plan de compra y puja y negocia en la misma. Los diversos compradores son sincronizados en tiempo real por un coordinador de de compras, que persigue el cumplimiento del plan de compra a nivel global. El responsable logístico de cada plataforma es el encargado de analizar la viabilidad de tener los pedidos (entregas y recepciones) disponibles, así como de sincronizar con el responsable de transporte la llegada y salida de los camiones y furgonetas. La figura 8.1 muestra un resumen gráfico del contexto del caso de estudio.

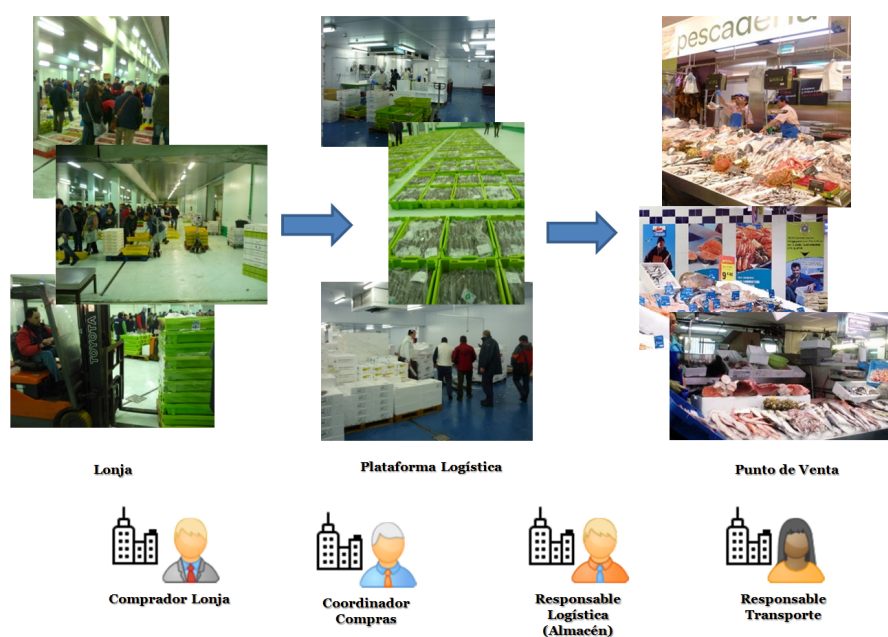


Figura 8.1: Contexto general del modelado del proceso colaborativo de compra y distribución de pescado fresco

Dentro del proyecto, era necesario modelar el proceso de compra y distribución, partiendo de lo que cada uno de los participantes realiza. Encontramos que algunos procesos estaban no formalmente establecidos (ej. cómo el comprador en la lonja toma las decisiones) y otros atendían a modelos de calidad (ej. manipulación de pescado en la plataforma logística), por lo que el primer paso fue hacer una representación de los procesos a nivel individual y trabajar en reuniones con cada participante para modelar el proceso en un marco colaborativo. La figura 8.2 muestra un ejemplo del trabajo realizado para el caso del comprador en la lonja.

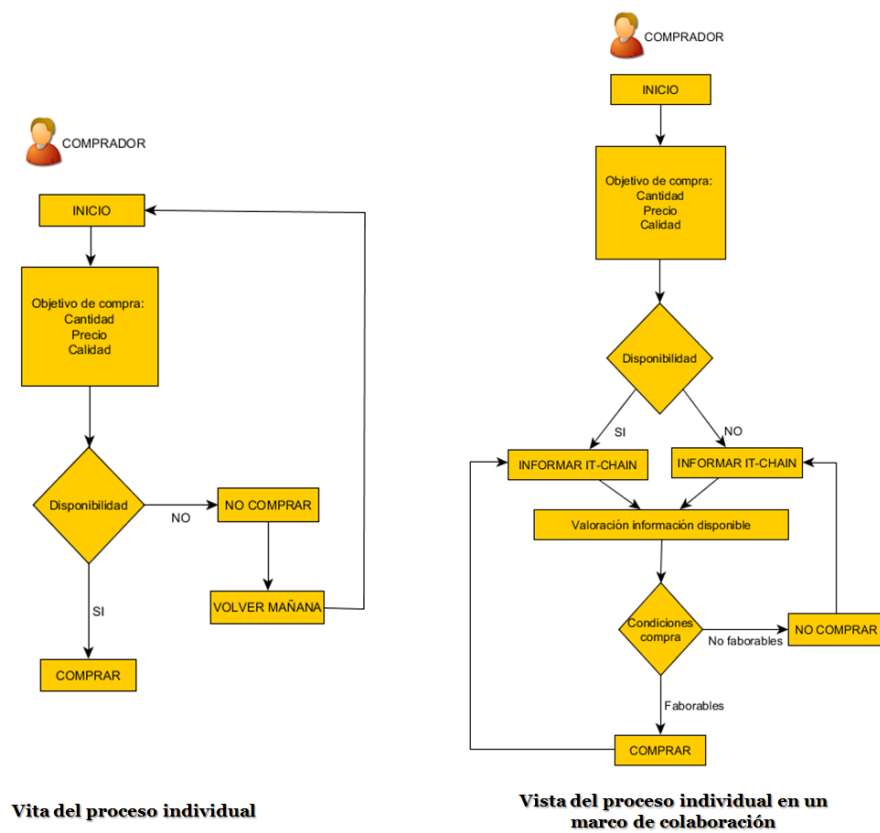


Figura 8.2: Vistas del modelo de proceso individual y en colaboración obtenidas en ITChain

A la vista de cada una de las vistas de los procesos individuales en un marco de colaboración, se representó un modelo gráfico del proceso colaborativo, tal y como muestra la figura 8.3.

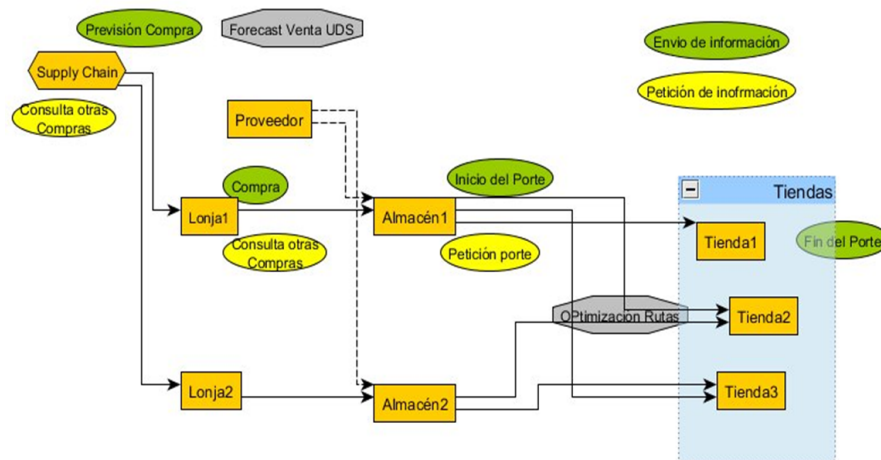


Figura 8.3: Modelo de proceso colaborativo obtenido en ITChain

Esta operativa para generar el modelo de proceso colaborativo evidenció los siguientes problemas:

- El modelo de proceso en colaboración no se basa en los procesos individuales que pudiera haber.
- Los modelos de procesos individuales existentes son traducidos e interpretados manualmente a una sintaxis gráfica común, pudiendo cometer errores y no garantizando la consistencia ante cambios futuros.
- No se garantiza una trazabilidad entre el modelo en colaboración y los individuales.
- Si un participante colabora en varias CdS, hay que repetir este proceso en cada una manualmente, por lo que las inconsistencias entre la vista individual y las colaborativas aumentan.

En definitiva, este proyecto evidenció que la generación de un modelo de proceso colaborativo requería de un método y de una solución que permitiera reutilizar los modelos de procesos individuales, de forma que el resultado sea un verdadero elemento de ayuda a la toma de decisiones a nivel práctico.

Seguidamente vamos a exponer cómo se hubiera realizado este modelado de proceso colaborativo atendiendo a la solución CBG planteada en este trabajo de tesis. Para ello, vamos a aplicar el método general CBG detallado en la figura 3.13.

Cumplimos la *Precondición 1 (PRE1)*, ya que tenemos un escenario de proceso colaborativo a partir de un conjunto de procesos individuales de diversas organizaciones.

8.2.1. Obtener la descripción de los procesos individuales

Desarrollamos la *Actividad 1 (A1)*, en la que se parte de los modelos de procesos individuales definidos en cualquier lenguaje y atendiendo a la solución planteada por García-Borgoñón [García-Borgoñón, 2016] se obtiene una descripción unificada de los procesos con el lenguaje INROMA. Dado que en el proyecto no había una descripción formal de los procesos, hemos generado directamente el modelo INROMA. En el caso de que alguno de los procesos estuviera modelado, por ejemplo con BPMN, IDEF o diagramas de actividad UML, a través de la solución MONETA [García-Borgoñón, 2016] se podría generar de forma automática el modelo INROMA. Con cualquier otro lenguaje, se podría añadir al marco de referencia planteado [García-Borgoñón, 2016]. De esta forma, disponemos de cuatro modelos de procesos individuales descritos en INROMA, que pasamos a detallar a continuación:

Proceso individual de compra en la lonja

El comprador en la lonja, identificado como un *actor* en el proceso, parte con un plan de compra, representado por un *deliverable* fruto del proceso de aprovisionamiento fuera del alcance de nuestro proyecto. En dicho plan se establece qué cantidad de qué referencia debe adquirir y una horquilla de precios. Como primera tarea debe negociar con los proveedores de la lonja la cantidad de producto disponible y el precio de la adquisición. Para cada referencia, se establece un código de producto sustitutivo, si bien el objetivo es intentar cumplir el plan de compra inicialmente establecido. Tras las negociaciones y, antes de proceder a confirmar la compra, el comprador efectúa una propuesta de pedido que es generada hacia el coordinador nacional.

Es a través de ese *producto*, como se lleva a cabo la coordinación entre el proceso del comprador en la lonja con el comprador nacional. A la vista de cómo ese producto se vea modificado o aprobado, el comprador nacional actuará en consecuencia. En el caso de que el coordinador apruebe el pedido de compra, creará el pedido oficial que será representado como un *deliverable*, y finalizará su trabajo. En otro caso, deberá volver a renegociar con los proveedores atendiendo al nuevo plan de compra recibido.

Este trabajo se realiza de forma iterativa con cada una de las referencias, de forma coordinada con el resto de los compradores en lonjas en tiempo real.

Las principales *métricas* del proceso a nivel individual son: 1) la cantidad que estaba prevista adquirir y la que se ha satisfecho finalmente, agregada en total y desagregada por referencia y fecha (el análisis se realiza por días, semanas, meses, años y los interanuales correspondientes), y; 2) el precio planificado del aprovisionamiento y el finalmente adquirido, de nuevo global y desagregado por unidad temporal y referencia.

El *indicador* de base establece tres niveles de cumplimiento: 1) verde: cuando el grado de cumplimiento de la planificación es mayor al 80 %; 2) amarillo: cuando el cumplimiento está entre el 40 % y el 80 %, y; 3) rojo: cuando no se alcanza el 40 % de cumplimiento.

Cabe destacar que este proceso individual sería uno de los que podrían entrar en los denominados procesos intensivos en conocimiento, ya que la actividad de negociar precio depende de un conjunto de circunstancias en las que el conocimiento de la persona específica que realiza la compra es más determinante que la secuencia de acciones definida en el proceso. Por ello, las desviaciones pueden estar perfectamente justificadas (se prevé mala mar y durante varios días el producto no va a estar disponible y es preferible adquirir hoy más producto que el planificado), si bien el comprador no tiene visibilidad de los efectos que puede tener esa decisión en el resto de la CdS. Se optimiza el proceso individual, pero no se sabe qué efecto tiene en la CdS. Si compra mucho más, hará falta más personal a nivel logístico, es posible llenar camión y medio, por lo que hay que buscar mercancía para llenar ese otro medio camión, etc. Situaciones que hemos podido evidenciar en la realidad y que motivaron el proyecto en particular.

El modelo de proceso individual de la compra en la lonja se expone en la figura 8.4.

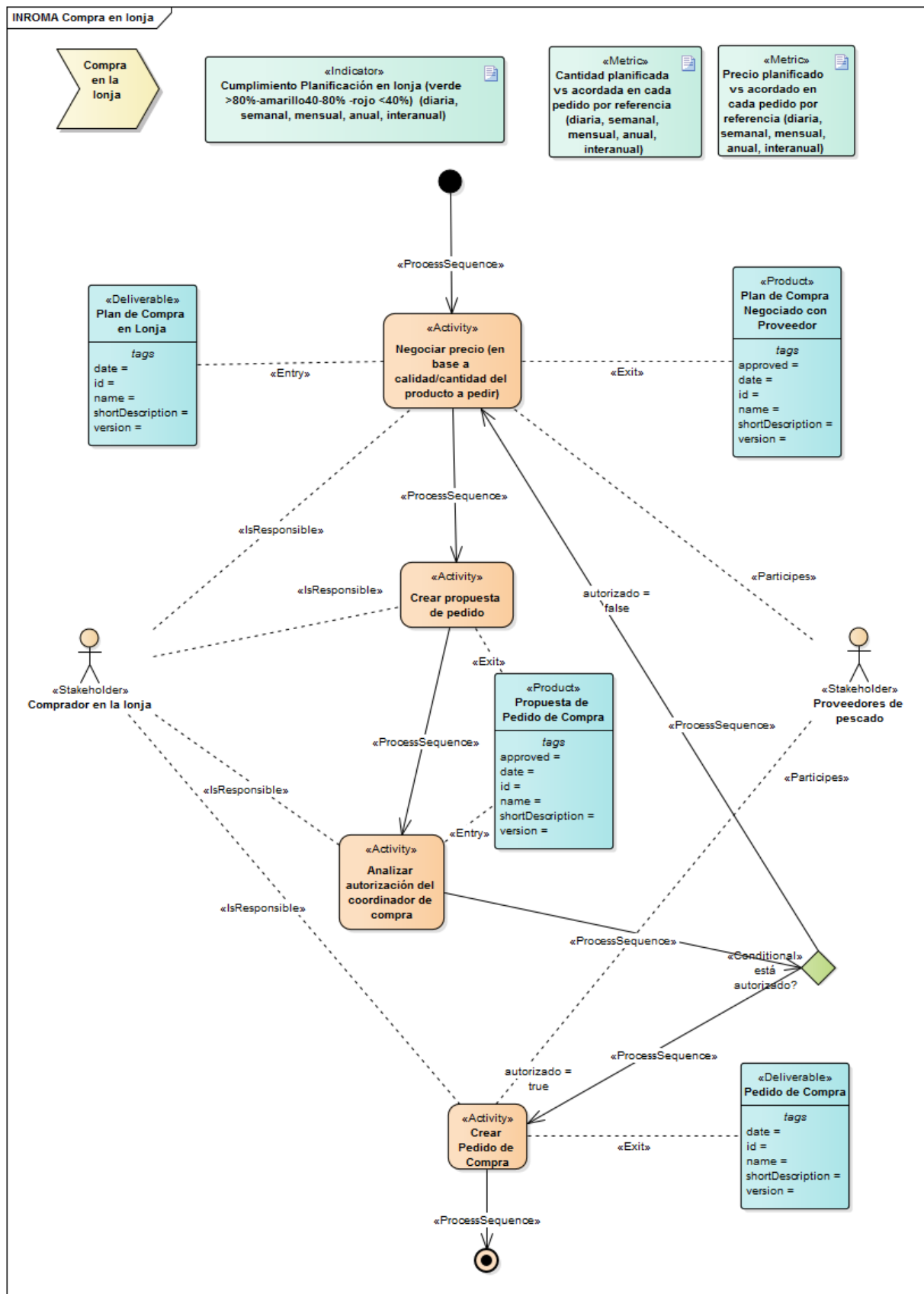


Figura 8.4: Modelo de proceso individual en INROMA de la compra en la lonja

Proceso individual de coordinación de compras en las lonjas

El coordinador de compras recibe las diversas propuestas de compra de los responsables en cada una de las lonjas, y lo compara con el plan que se había obtenido en el proceso de aprovisionamiento. En el caso de que la propuesta de pedido se corresponda con lo planificado, realiza la autorización. En caso contrario, es preciso que el operador logístico los autorice, ya que se ha dimensionado el personal y las ventanas temporales de recogida y entrega de pedidos al plan de compra. En el caso de que las compras sean mayores, el responsable del almacén en el que se prepara y manipula la mercancía, en este caso además como producto perecedero a temperatura controlada, es necesario garantizar que existirá personal para gestionar esa desviación y poder realizar las entregas a tiempo.

En este punto el coordinador de compras tampoco tiene una visibilidad global de la CdS, tan sólo sabe que si hay desviaciones debe notificarlas al operador logístico y condiciona la aprobación al comprador de la lonja a que el responsable de almacén haga lo propio.

Las principales *métricas* del proceso a nivel individual son: 1) la cantidad que estaba prevista adquirir y la que se ha satisfecho finalmente por todos los compradores de lonja, agregada en total y desagregada por referencia y fecha (el análisis se realiza por días, semanas, meses, años y los interanuales correspondientes), y; 2) el precio planificado del aprovisionamiento y el finalmente adquirido, de nuevo global y desagregado por unidad temporal y referencia.

El *indicador* de base establece tres niveles de cumplimiento: 1) verde: cuando el grado de cumplimiento de la planificación es mayor al 80 %; 2) amarillo: cuando el cumplimiento está entre el 40 % y el 80 %, y; 3) rojo: cuando no se alcanza el 40 % de cumplimiento.

El modelo de proceso individual de la coordinación de la compra se expone en la figura 8.5.

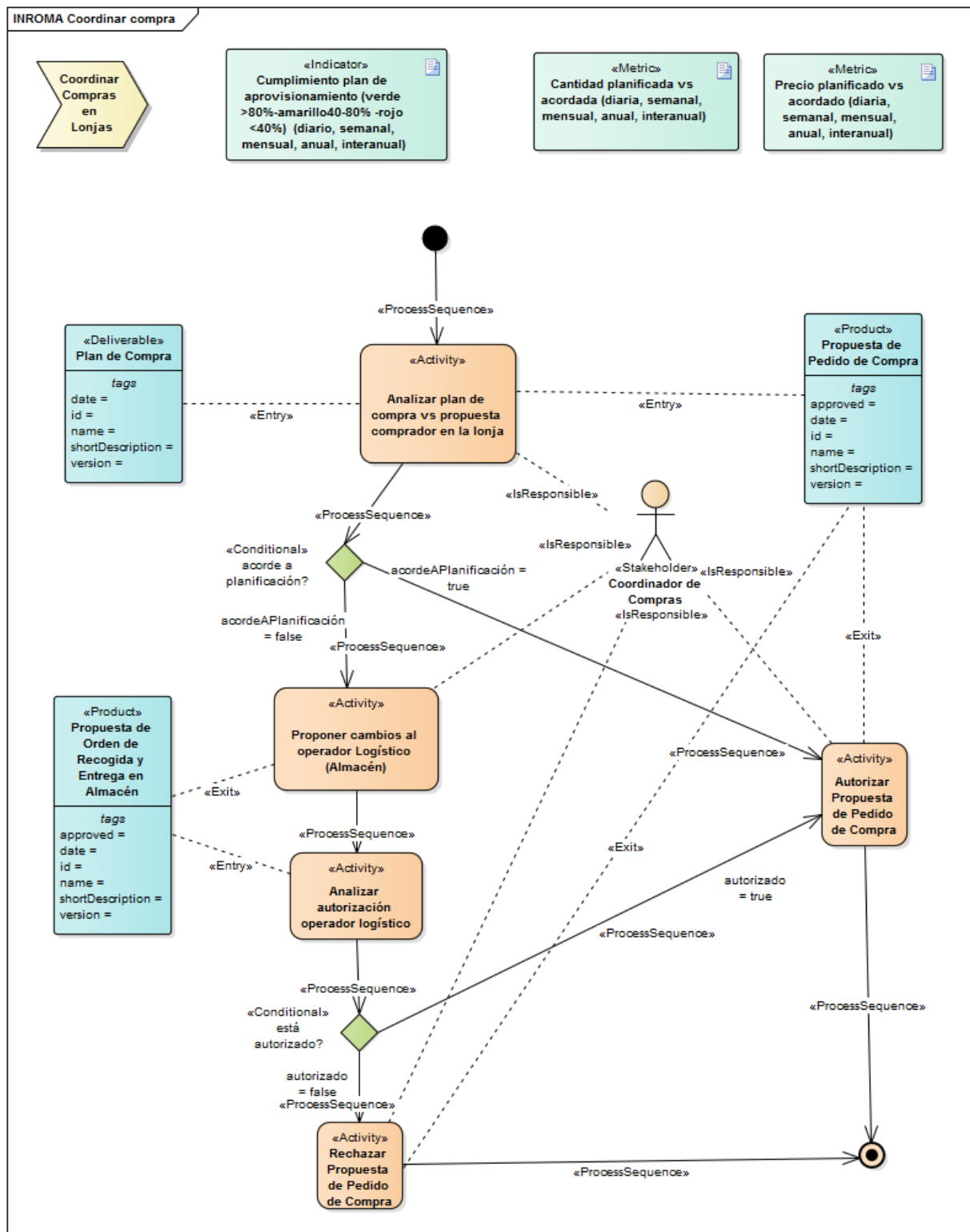


Figura 8.5: Modelo de proceso individual en INROMA de la coordinación de la compra

Proceso individual de gestionar cambios en la planificación logística del almacén

El operador logístico es el responsable de gestionar la recogida del producto, manipularlo y preparar los pedidos que serán suministrados a otras plataformas logísticas y a los puntos de venta, garantizando que no se rompe la cadena de frío. En este caso existen diversos operadores, si bien en el proyecto se generalizó a un mismo modo de operación, siendo conscientes de que la realidad es más compleja.

El operador ha dimensionado su equipo y las ventanas temporales para la recogida, manipulación y expedición de los pedidos. En un flujo tenso, existe poco margen de maniobra desde que se recibe el pescado hasta que los camiones llegan a la hora acordada para su recogida. En el caso de que el plan de compra se modifique, se debe analizar qué impacto tiene esa nueva compra en el plan establecido, de cara a analizar la viabilidad.

Si no es viable, directamente se rechazará, siendo la imposibilidad de cumplir las ventanas temporales la causa más habitual. Si lo es, lo normal será que quizás los horarios establecidos deban ser modificados. Es decir, el operador puede dimensionar un nuevo equipo de personas para que manipule la mercancía, pero quizás el producto estará disponible con unos minutos u horas de retraso. Esa nueva ventana temporal, a la vista del operador puede ser válida, pero sabe que en la CdS hay otro participante responsable de garantizar que el producto llega al destino a tiempo. Por ello, antes de confirmar, debe plantear el nuevo escenario al operador de transporte que, a su vez, validará la nueva propuesta para que esa confirmación viaje aguas abajo de nuevo en la CdS.

Como vamos viendo en cada una de las vistas, un actor de la CdS es consciente del eslabón más cercano, pero no tiene una visibilidad global sobre los efectos que una decisión suya puede tener en el resto de la CdS.

El modelo de proceso individual de gestionar cambios en la planificación logística del almacén se expone en la figura 8.6.

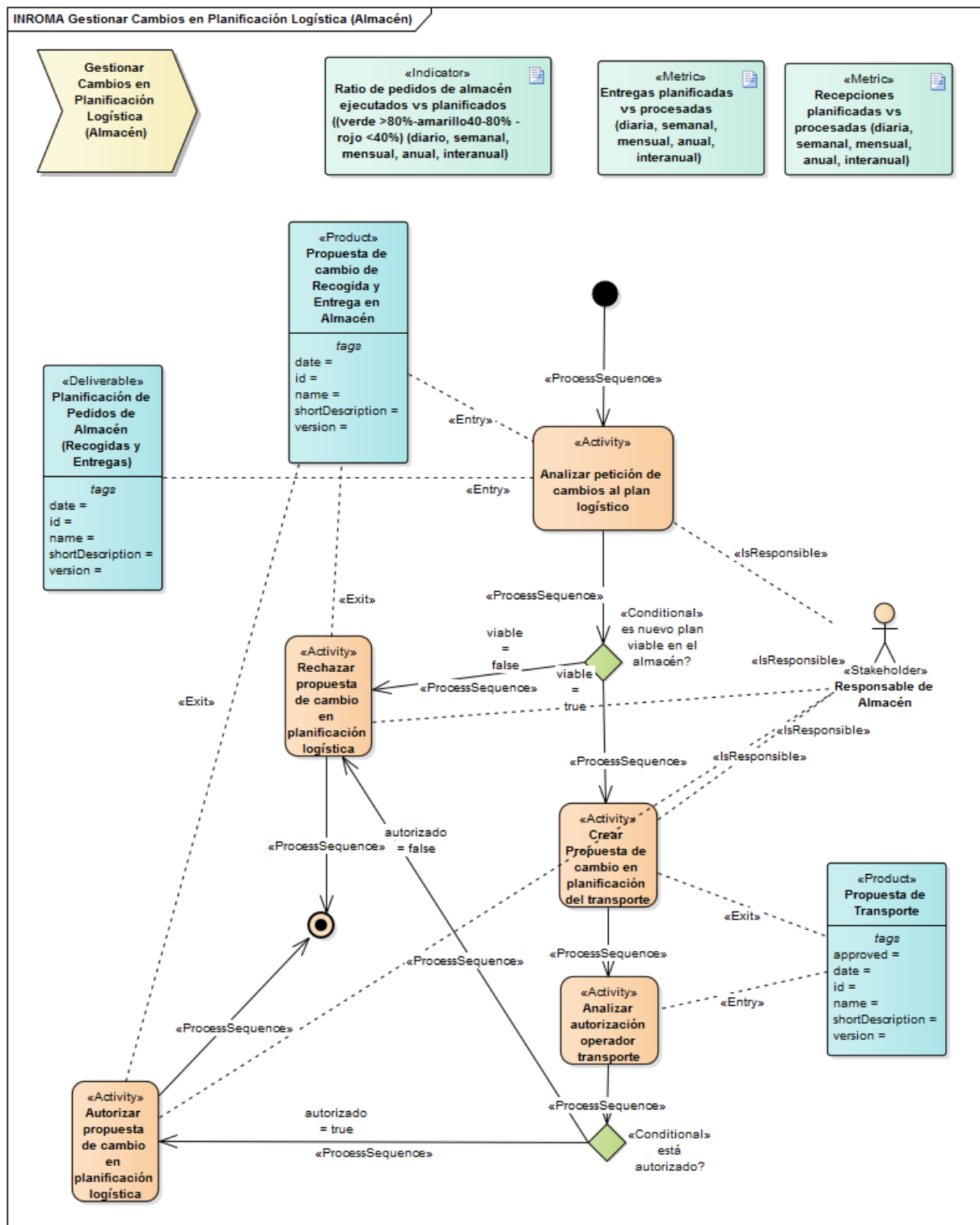


Figura 8.6: Modelo de proceso individual en INROMA de un cambio en la planificación logística (almacén)

Proceso individual de gestionar cambios en la planificación del transporte

Finalmente, para terminar de ilustrar la CdS tenemos al operador de transporte. De nuevo la realidad nos lleva a múltiples escenarios, desde compañías globales hasta personal autónomo. Desde rutas fijas con origen la plataforma logística y destino algún centro comercial de gran tamaño, hasta rutas inter urbanas que cubren pequeños centros de proximidad, o rutas entre plataformas logísticas en las que se consolida la mercancía junto con otras para hacer el reparto final al punto de venta. Al igual que con el operador logístico, en el proyecto se hizo la hipótesis de representar un modelo de comportamiento homogéneo.

El operador de transporte recibe el impacto que puede tener un cambio respecto al plan inicialmente establecido, derivado de una nueva ventana temporal de entrega y recogida (si hay más mercancía posiblemente habrá que recogerla más tarde en la plataforma pero quizás la tienda requiera una entrega más temprana si no disponen de más personal para ubicar todo el producto en el lineal). En el caso de que sea factible cubrir los nuevos horarios establecidos, se aprobará, volviendo el bucle aguas abajo para validar el resto de cambios. En otro caso, se rechazará.

Un escenario posible es que los cambios en el transporte no vengan simplemente por nuevas ventanas temporales sino por mayor necesidad de espacio. Lo que antes cabía en una furgoneta, ahora requiere de un camión. Lo que antes llenaba un camión, ahora puede necesitar de un camión y medio. Transportar aire es muy costoso en la industria logística, por lo que en esos escenarios lo habitual es que incluso cuando el operador logístico de transporte pueda acometerlos, se busque en coordinación con el resto de la cadena quién puede adelantar el envío de mercancía para llenar ese otro posible medio camión.

Como vemos, estos escenarios reales nos ofrecen un claro ejemplo de cómo las decisiones de un actor condicionan y están cada vez más condicionados con el resto con los que comparte una cadena de valor.

El modelo de proceso individual de gestionar cambios en la planificación del transporte se expone en la figura 8.7.

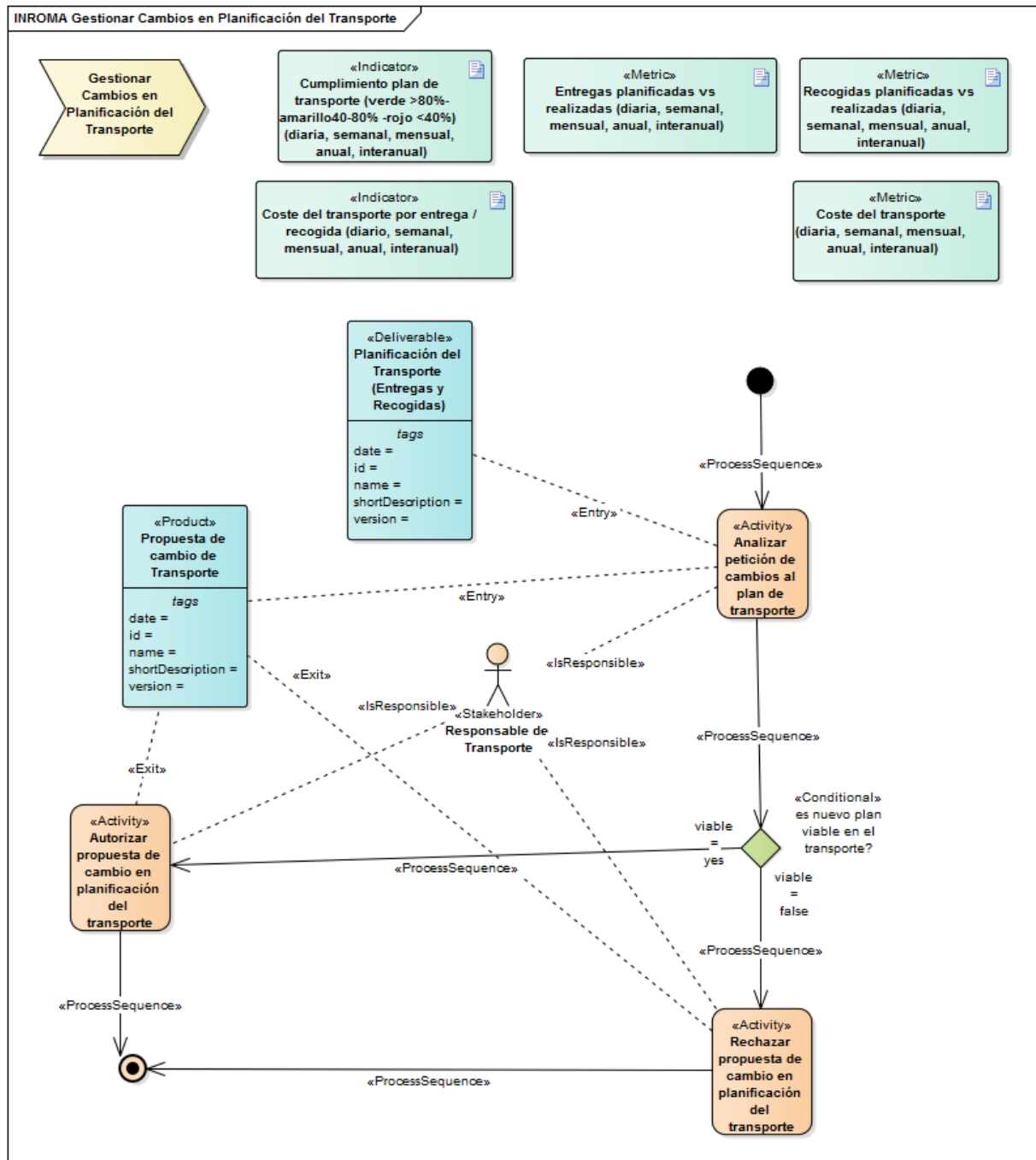


Figura 8.7: Modelo de proceso individual en INROMA de un cambio en la planificación del transporte

8.2.2. Establecer la vista colaborativa del proceso individual

A partir de la descripción INROMA de los procesos individuales, a través de la *Actividad 2 (A2)*, se genera qué parte del proceso formará parte de la colaboración, es decir, qué elementos quiere cada participante hacer públicos y cuáles mantener bajo privacidad y autonomía. Como resultado tenemos un modelo, descrito conforme a *CBGProcess*, en el que se establece la vista pública de la colaboración.

En este caso haremos uso de la solución CBG-Tool para de forma automática, a través de la transformación horizontal definida en la sección 6.2.1, tener una primera versión de la vista individual de los procesos de cada organización en la colaboración. Siguiendo el método definido en la figura 4.3 completaremos la vista individual de cada proceso organizativo, a través de las siguientes acciones:

- A1: Etiquetando las organizaciones.
- A2: Identificando los productos colaborativos.
- A3: Identificando las actividades colaborativas.
- A4: Definiendo el flujo de proceso en colaboración.
- A5. Identificando las métricas e indicadores del proceso en colaboración.

Como resultado tendremos un modelo de proceso colaborativo definido a partir de los procesos individuales de las organizaciones, que pasamos a exponer a continuación.

Vista colaborativa del proceso de compra en la lonja

Reutilizando el modelo individual del proceso, el comprador en la lonja establece, conforme al metamodelo *CBGProcess*, qué elementos del proceso son públicos y cuales privados. En este caso vemos que las actividades propias del proceso se mantienen como privadas, son actividades individuales marcadas en color rojo. Para indicar la colaboración, se ha establecido la tarea coordinar compra colaborativa, que contiene la actividad interna crear propuesta de pedido. El producto propuesta de pedido de compra, existente en la vista individual, en este caso sí que se marca como público, como mecanismo de intercambio de información entre el comprador de la lonja y el coordinador. El primero le dirá lo que puede comprar, el segundo lo validará.

De esta forma tenemos un proceso lo más parecido al que tenía el comprador, porque hemos partido del mismo, pero extendido para aquellos elementos que el comprador de la lonja requiere de la colaboración con el resto de la CdS. Nuestro enfoque nos permite tener privacidad (el comprador decide en este caso que la actividad se mantiene privada, haciéndola pública sólo una parte de la misma, mientras que el producto es público), y la autonomía, ya que es perfectamente consciente de que las actividades individuales son plenamente autónomas respecto al resto de la CdS.

El modelo con la vista colaborativa del proceso de compra en la lonja se expone en la figura 8.8.

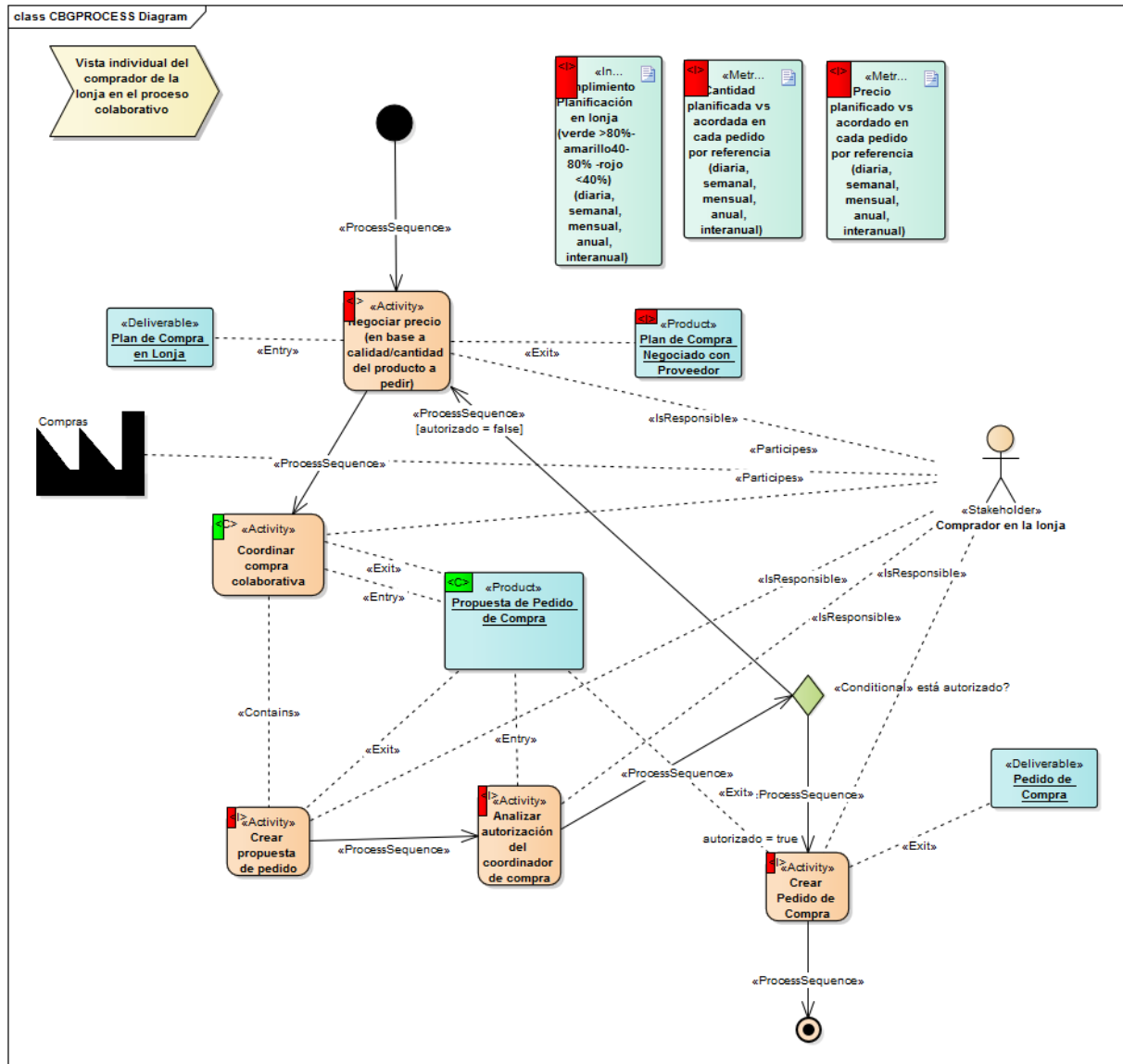


Figura 8.8: Modelo de la vista individual del proceso colaborativo desde el comprador en la lonja

Vista colaborativa del proceso de coordinación de compras en las lonjas

De la misma forma procedemos con el proceso dirigido por el coordinador de compras. En este caso algunas de las actividades (analizar plan de compra, proponer cambios al operador logístico, analizar autorización operador logístico y rechazar propuesta de pedido de compra) se mantienen autónomas y privadas, marcadas en rojo. Otras, pasan a ser visibles a toda la CdS (autorizar propuesta de pedido), mientras que existe un tercer caso de que se hace pública una parte de la colaboración (coordinar disponibilidad logística en el almacén de forma colaborativa).

De esta forma podemos evidenciar todos los mecanismos de trazabilidad entre la vista individual y la colaborativa que el enfoque bottom-up del marco CBG nos ofrece: mantenerlo privado (*BlackBox Activity*), hacerlos públicos mediante la extensión (*Activity*), o bien hacer una nueva actividad pública que contiene aquellas privadas que no queremos hacer visibles al resto de la CdS.

De la misma forma tendremos productos, métricas e indicadores, tanto públicos como privados. Poco a poco vamos evidenciando el potencial del marco CBG a la hora de crear el modelo de la colaboración reutilizando los modelos de procesos individuales. El objetivo es que el comprador de la lonja tenga su proceso interno definido, sea consciente de en qué puntos le puede afectar o puede condicionar la colaboración con el resto de la CdS, pero que no pierda su privacidad ni su autonomía, algo clave para mantener su competitividad y su éxito empresarial.

El modelo con la vista colaborativa del proceso de coordinación de compras en las lonjas se expone en la figura 8.9.

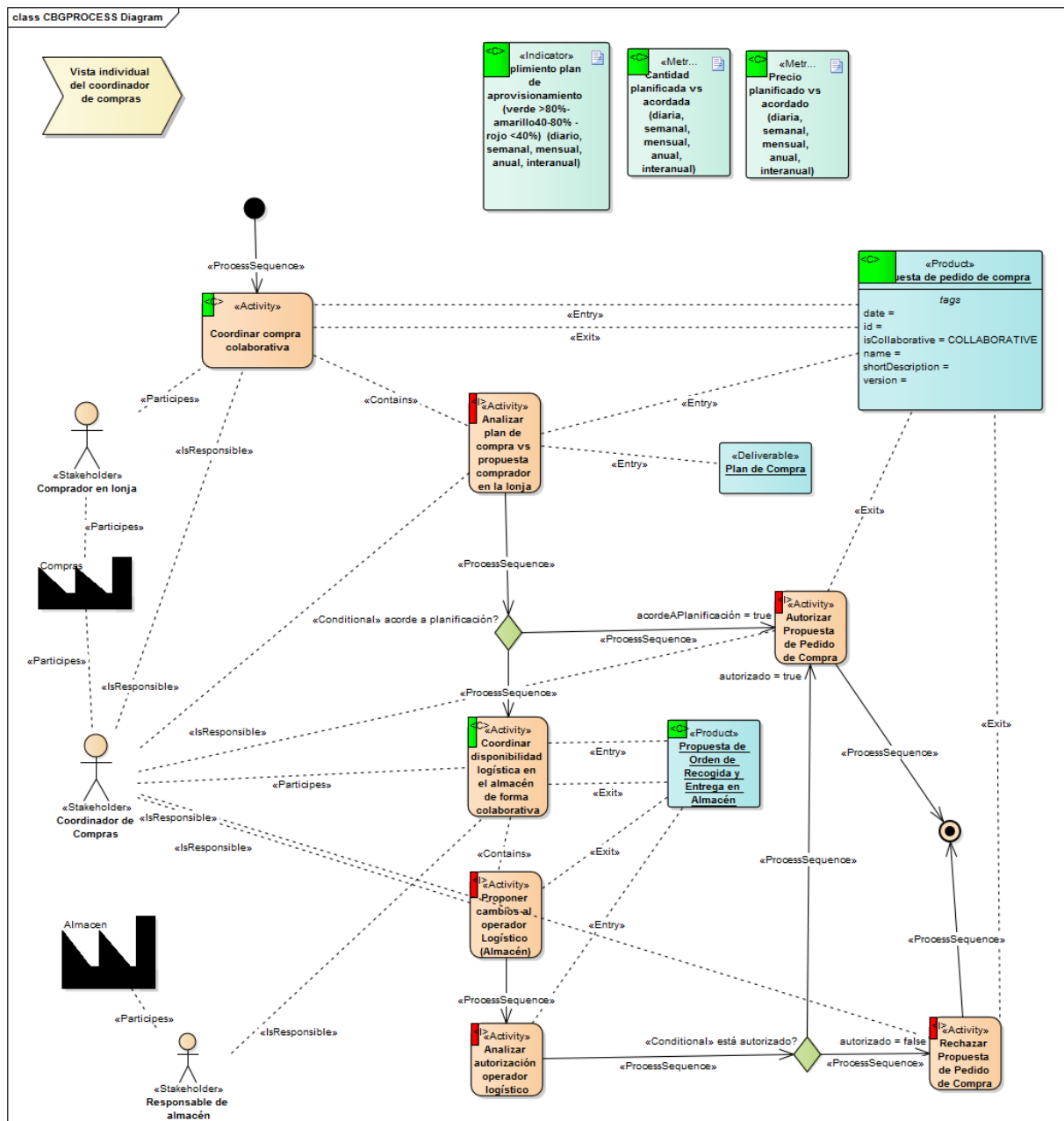


Figura 8.9: Modelo de la vista individual del proceso colaborativo desde el coordinador de compras

Vista colaborativa del proceso de gestionar cambios en la planificación del transporte

El modelo con la vista colaborativa del proceso de gestionar cambios en la planificación del transporte se expone en la figura 8.11.

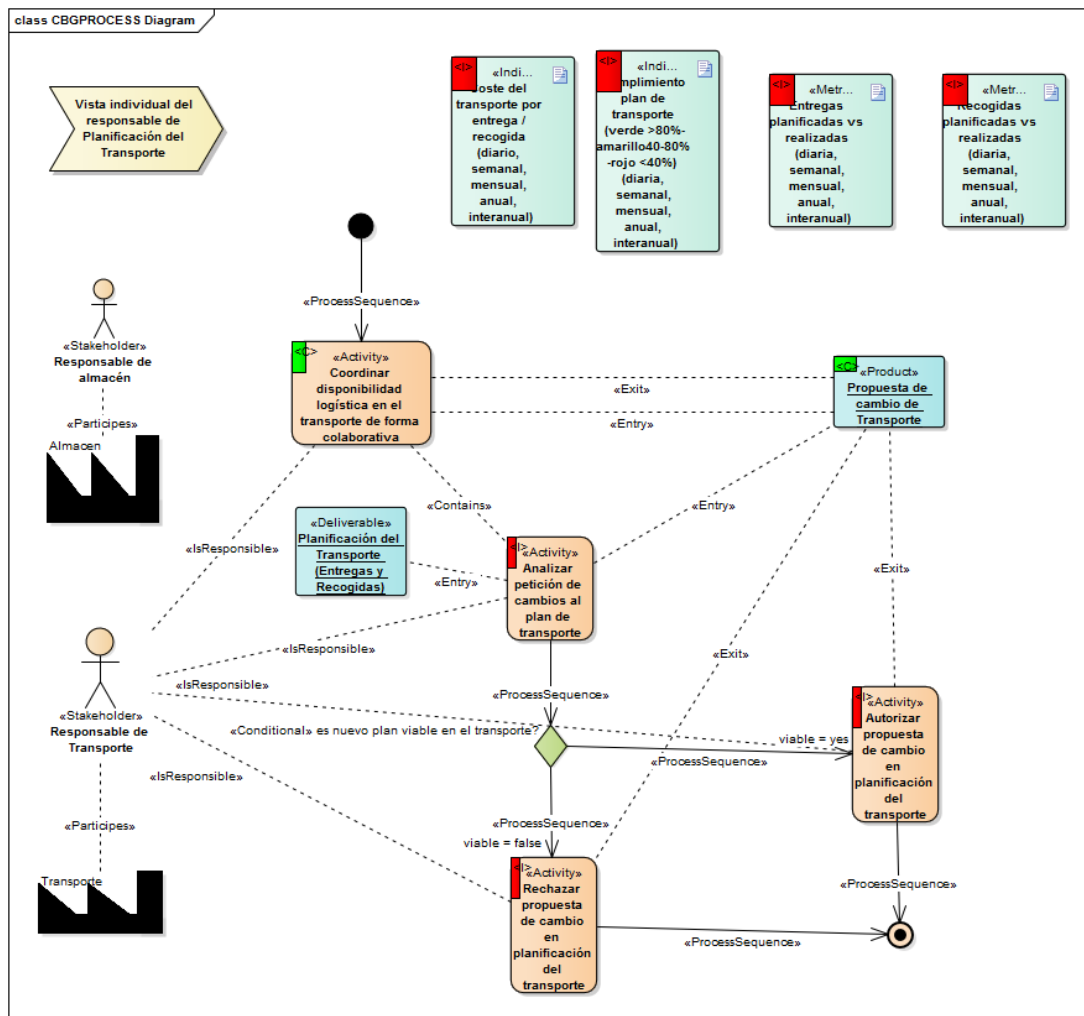


Figura 8.11: Modelo de la vista individual del proceso colaborativo desde el responsable de transporte

8.2.3. Construir el modelo de proceso colaborativo

Una vez tenemos la vista colaborativa de cada uno de los procesos individuales, mediante la *Actividad 3 (A3)* del método, se genera el *Modelo de proceso colaborativo*, primer resultado del proceso.

Para ello procederemos de igual forma que en el punto anterior cuando hemos llevado a cabo la vista individual de las organizaciones, siguiendo el método definido en la figura 4.3, es decir:

- A1: Añadiremos las organizaciones que forman parte de la colaboración.
- A2: Identificamos los productos colaborativos.
- A3: Creamos las actividades colaborativas.
- A4: Definimos el flujo de proceso en colaboración.
- A5. Establecemos las métricas e indicadores del proceso en colaboración.

Como resultado, obtenemos el modelo de proceso colaborativo representado en la figura 8.12, correspondiente a la Postcondición 1 (POST1): modelo de proceso colaborativo definido a partir de los procesos individuales de las organizaciones.

En dicho modelo podemos observar que es necesario representar a todas las organizaciones que forman parte de la CdS, lo que nos aporta una clara visibilidad de quiénes formamos la cadena de valor. También están definidos los diversos stakeholders de cada organización y de qué forma se colabora, al establecer quién es responsable y quién participa en cada actividad.

En el modelo colaborativo todas las actividades serán en colaboración, ya que aquellas individuales han quedado protegidas de forma privada en los modelos de cada organización que, si bien han servido para generar la vista de la colaboración, se mantienen de forma autónoma y no son visibles al resto de participantes.

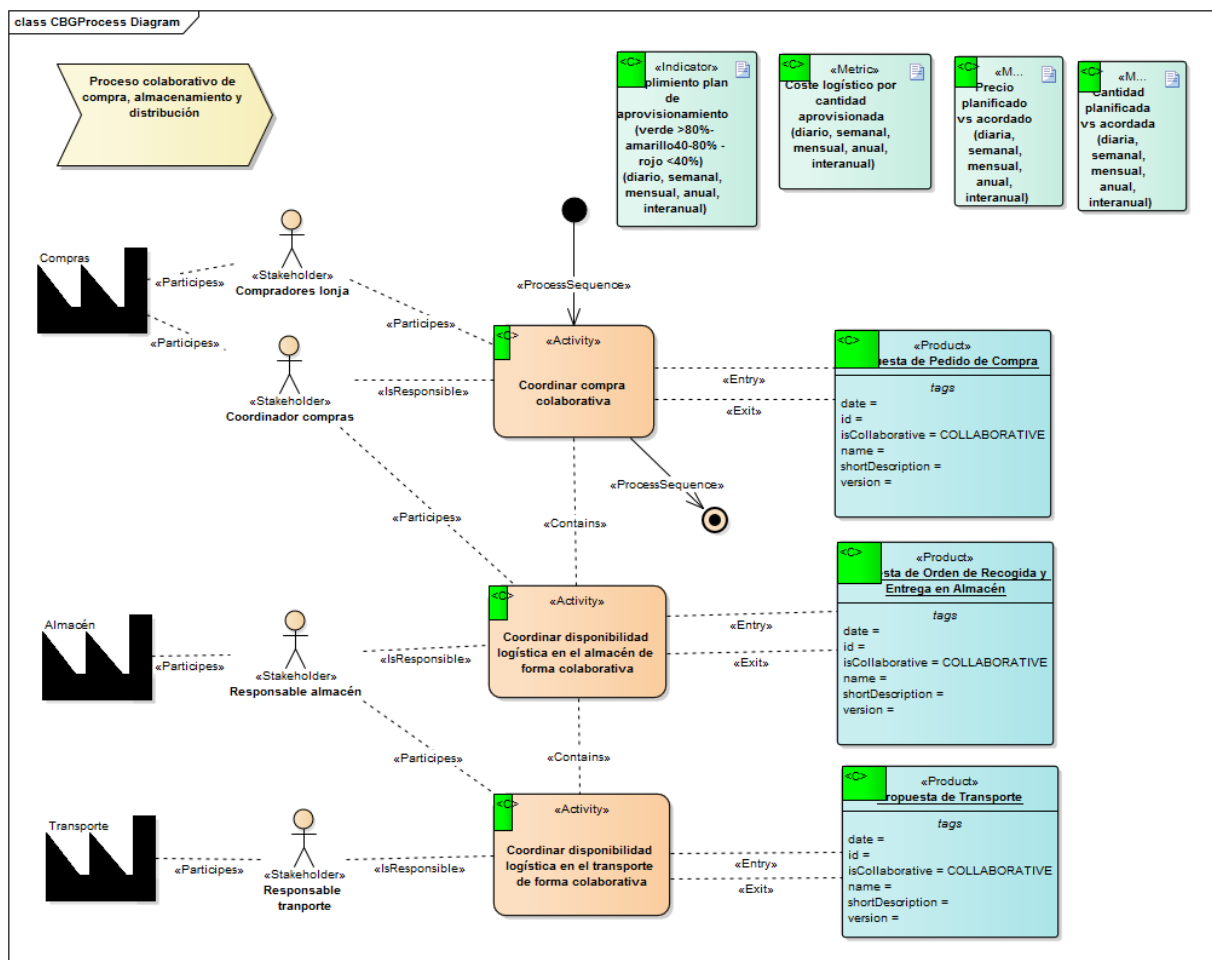


Figura 8.12: Modelo del proceso colaborativo de aprovisionamiento, transporte y distribución

8.2.4. Construir el modelo de caso colaborativo

Si en el dominio de aplicación no es requerida la parte dinámica, en cuanto a la planificación de actividades, el proceso finaliza. En otro caso, se realiza una conversión del dominio de procesos al de casos, de forma que las actividades *A4* y *A5* resumen el método para generar el caso colaborativo, que será detallado en la sección 5.3.

- Actividad 4 (A4): A partir del proceso colaborativo se realiza una transformación entre modelos que genera un primer caso colaborativo, que incluye la posibilidad de establecer qué elementos del proceso son susceptibles de ser planificados en tiempo de ejecución.
- Actividad 5 (A5): Por último, atendiendo al metamodelo de casos colaborativos, se finaliza la descripción del *Modelo de caso colaborativo*, segundo *Deliverable*.

Como resultado, dispondremos de la Postcondición 2 (POST2): modelo de caso colaborativo definido a partir del proceso colaborativo anteriormente definido, mostrado en la figura 8.13.

Para ilustrar el caso, hemos extendido la visión del proceso colaborativo para dar representar no sólo la fase de la compra, sino incluir las tareas relativas a su procesado en los almacenes (distribución) y su transporte, así como las posibles incidencias que puedan acontecer hasta su llegada al punto de venta.

De esta forma podemos comprobar que:

- El caso comienza con una secuencia obligatoria en la que se debe coordinar la compra de forma colaborativa. Esta tarea incluirá la coordinación con la parte de almacén que a su vez incluye la disponibilidad logística en el transporte.
- Cerrada la compra, se establece una secuencia obligatoria hacia la coordinación de la distribución, momento en el que la mercancía adquirida es llevada a los almacenes para su procesamiento. Cuando toda la mercancía está preparada, queda disponible para su transporte, algo que se representa a través de una *Constraint* que la primera tarea genera y la siguiente aguarda.
- Tras esa actividad se establece una secuencia opcional hacia el transporte, que requerirá de una planificación dinámica establecida a través del flujo *Plan*. De esta forma representamos que el coordinador de transporte podrá ir combinando sus tareas individuales dentro del caso para establecer las mejores asignaciones a flotas y rutas posibles.
- Finalizado el transporte de nuevo se establece una secuencia opcional con capacidad de re-planificación dinámica para atender las posibles mermas, roturas o incidencias acontecidas en la entrega a los puntos de venta.

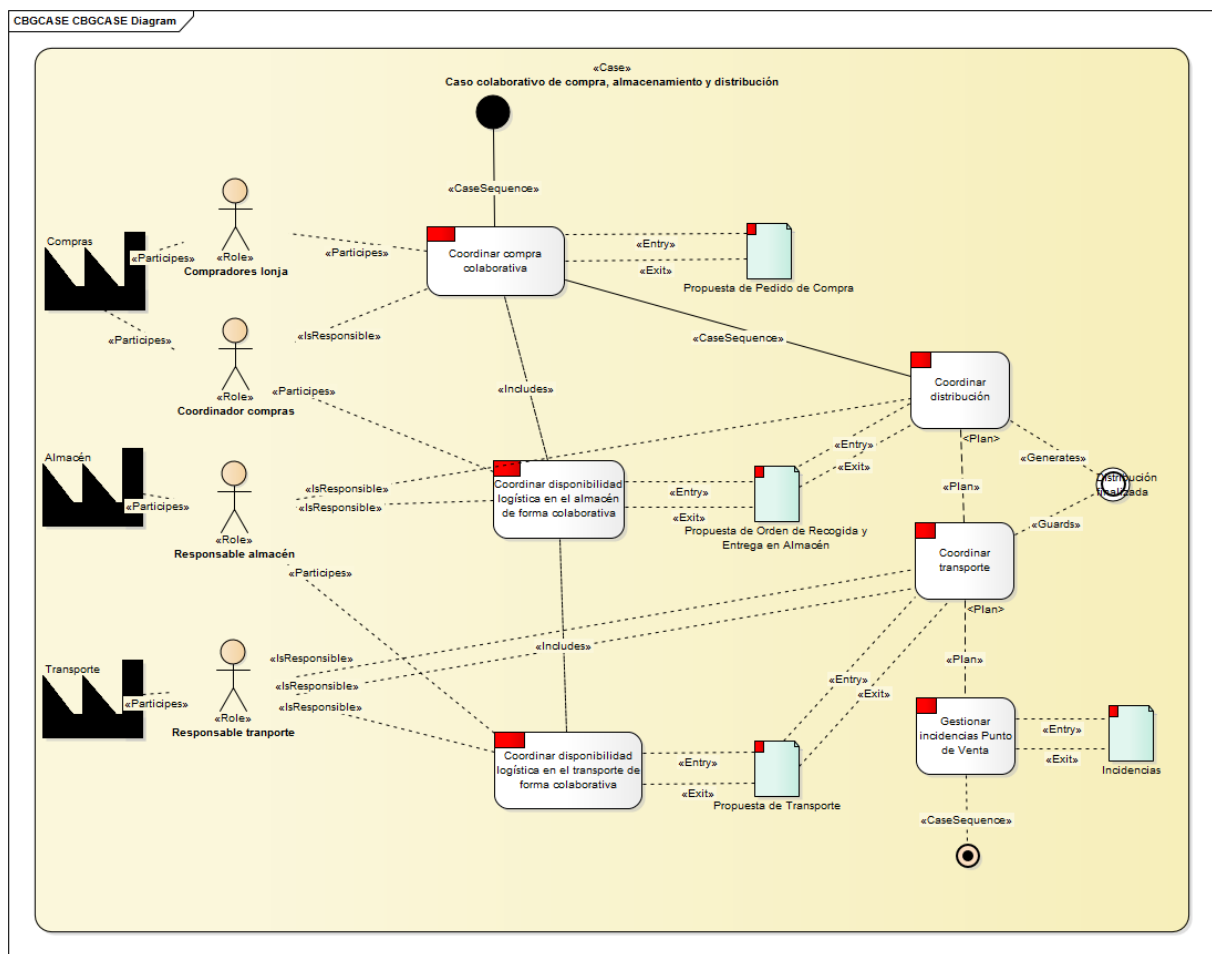


Figura 8.13: Modelo del caso colaborativo de aprovisionamiento, transporte y distribución

8.3. Conclusiones

En este capítulo se ha presentado el caso de estudio real del proceso de compra y distribución de pescado fresco, desde las lonjas hasta el punto de venta, pasando por unas plataformas logísticas de manipulación y almacenaje del producto dispersas por todo el territorio nacional, de la organización CARREFOUR en España.

Siguiendo los métodos definidos y haciendo uso de la solución CBG-Tool, hemos ido generando la vista individual de cada proceso, su visión particular dentro de la colaboración así como el modelo de proceso colaborativo. En este sentido hemos podido validar que nuestro enfoque permite:

- Generar la vista colaborativa a partir de los modelos de procesos individuales.
- Disponer de una vista individual y particular de cada organización dentro de la colaboración, manteniendo su privacidad en el proceso y su autonomía a la hora de tomar decisiones.
- Tener una trazabilidad entre las actividades y productos en colaboración con la vista individual de cada organización, de forma que se pueda dotar de flexibilidad a la misma en cuanto a su posible participación en varias CdS.

Por último, continuando con el método propuesto y haciendo uso de las transformaciones de procesos a casos dentro de la herramienta CBG-Tool, hemos validado cómo se podría incluir la capacidad de dinamismo para replanificar en tiempo de ejecución las posibles tareas, generando un modelo de caso colaborativo. En este aspecto hemos podido validar que nuestro enfoque permite:

- Incorporar dinamismo a partir del modelo de proceso colaborativo, generando de forma automática su equivalente en el dominio de casos.
- Mantener la trazabilidad desde el modelo de caso hacia atrás, de forma que podamos en tiempo de ejecución replanificar tareas del proceso colaborativo y, a su vez, cada organización de la CdS pueda identificar aquellas tareas que sólo forman parte de su vista individual de la colaboración.

De forma general, esta validación con un proceso complejo real nos ha permitido evidenciar que el marco CBG propuesto en este trabajo de tesis permite obtener una vista fiable de cómo se lleva a cabo un proceso en colaboración, en menos tiempo, con menos errores y facilitando la mantenibilidad de los modelos generados al poder incluir las características de flexibilidad y dinamismo requeridas en el dominio de la CdS.

Capítulo 9

Conclusiones y trabajos futuros

Durante este trabajo de tesis hemos ido abordando en profundidad uno por uno todos los elementos que conforman el marco de referencia CBG para dar soporte al modelado de procesos y casos colaborativos con un enfoque bottom-up, incorporando los aspectos de flexibilidad y dinamismo necesarios para dar soporte a una gestión de procesos colaborativos entre organizaciones. Para poder llevarlo a cabo, se ha realizado un estudio exhaustivo de la situación actual y, una vez analizados los resultados obtenidos, se han concretado las motivaciones que han cimentado el desarrollo de dicho marco.

Teniendo en cuenta que para resolver el problema planteado se ha optado por una perspectiva de ingeniería dirigida por modelos, nuestra propuesta de marco de referencia está soportado por los siguientes elementos claves que han constituido el grueso del trabajo aquí presentado:

- Un metamodelo, denominado *CBGProcess*, para dar soporte al modelado de procesos de negocio colaborativos.
- Una transformación M2M para generar la vista colaborativa desde un modelo de proceso individual.
- Un método para, a partir de la vista individual de un proceso conforme a *CBGProcess*, definir el proceso colaborativo. Dicho método permite separar la vista individual de un proceso de una organización de la vista colaborativa, manteniendo por tanto la privacidad y la autonomía en la toma de decisiones y compartiendo aquellos elementos que cada organización quiera para poder conformar el proceso en colaboración.
- Un metamodelo, denominado *CBGCase*, que siguiendo el paradigma de gestión de casos soporta el modelado de casos en entornos colaborativos.
- Una transformación M2M que permite, a partir de un proceso colaborativo conforme a *CBGProcess*, generar automáticamente una base de modelo de casos colaborativos conforme a *CBGCase*. De esta forma se consigue añadir la componente dinámica a los procesos, realizando una conversión del dominio de procesos al dominio de casos.

- Un método para, a partir del modelo de casos colaborativos generado, especificar qué elementos forman parte de la vista estática y cuáles en la vista dinámica del caso. De esta forma dispondremos del modelo de caso colaborativo generado a partir del proceso colaborativo.
- Una herramienta de soporte para facilitar la creación de modelos y la ejecución de las transformaciones desde un punto de vista práctico.

Además, para hacer viable el uso de esta solución teórica en entornos empresariales, se ha desarrollado la herramienta CBG-Tool, que da soporte práctico a todo este trabajo. Hemos validado la herramienta con un doble enfoque: por un lado, a partir de las situaciones reales inicialmente descritas en el planteamiento del problema que justificaban la realización de este trabajo de tesis y, por otro lado, se ha evaluado mediante la validación con el caso real de la colaboración en el aprovisionamiento, distribución y transporte de pescado fresco desde la lonja hasta el punto de venta de la organización CARREFOUR en España.

En este capítulo final queremos resumir cuáles han sido las principales aportaciones de este trabajo, así como las nuevas líneas de investigación que se abren y que a buen seguro serán cubiertas a corto plazo por nuevos doctorandos dentro del grupo IWT2.

Con tal fin, el capítulo se estructura como sigue. En primer lugar se presenta el contexto en el que se ha desarrollado esta tesis doctoral, que ha constituido una influencia importante en los resultados obtenidos. En la segunda sección se detallan las aportaciones de este trabajo, alineadas a los objetivos inicialmente planteados. Para finalizar, se proponen un conjunto de trabajos y líneas de investigación que siguen la senda iniciada por ésta y otras tesis doctorales vinculadas. La última sección resume las principales conclusiones obtenidas a lo largo del capítulo.

9.1. Marco de investigación en el que se desarrolla este trabajo

Este trabajo de tesis se ha visto influenciado por el contexto en el que se ha desarrollado, la actividad del grupo IWT2 y de ITAINNOVA así como las relaciones de dicho grupo con otros grupos de investigación y empresas, tal y como se describe en detalle a continuación.

9.1.1. Línea de investigación en el grupo IWT2

El grupo IWT2 posee desde hace unos años una línea de investigación en la que se trabaja en cómo combinar de forma satisfactoria el paradigma de ingeniería dirigida por modelos con la gestión de procesos de negocio en múltiples dominios. El poder contar con una línea sobre procesos nos permite alinear las propuestas y los planteamientos a las necesidades que se van detectando, de forma que los resultados de un trabajo de investigación apoyan las hipótesis de los siguientes, abriendo camino y avanzando conjuntamente.

Dentro de los objetivos de esta línea de investigación se contempla la necesidad de mejorar la competitividad de las organizaciones mediante la mejora continua de sus procesos de negocio, facilitando una gestión eficaz y eficiente de los mismos, utilizando para ello el paradigma MDE.

En este sentido ya hemos comentado en varias ocasiones que nuestra propuesta se basa en el trabajo llevado a cabo por García-Borgoñón, creando un marco de referencia para facilitar la interoperabilidad y la mantenibilidad de los procesos de software [García-Borgoñón, 2016]. Su propuesta INROMA ha sido la base del marco CBG para garantizar la independencia del proceso modelado frente al lenguaje de modelado seleccionado a la hora de abordar proyectos interorganizacionales.

Este trabajo contribuye a dar el soporte colaborativo en la aplicación de la ingeniería dirigida por modelos a la gestión de procesos de negocio, dentro del conjunto de trabajos del grupo que, de forma combinada, permiten abordar un conjunto de necesidades reales y transferir las soluciones a las organizaciones. De forma particular, nuestra propuesta tiene vínculos con:

1. La solución para orquestar y ejecutar los procesos de negocio a partir de su modelado [García García, 2015], en el que se consigue reducir, e incluso eliminar, la brecha existente entre los procesos definidos en las organizaciones y lo que realmente ocurre en el día a día de las mismas, es decir, las diferencias entre los procesos definidos y los procesos actuales.
2. La solución para proporcionar a las organizaciones la capacidad de modernización a través de la extracción dinámica de la vista global de sus procesos de negocio a partir de la información inferida en bases de datos relacionales, estructuras de tablas, restricciones y activadores PL/SQL de reglas de negocio [Arévalo Maldonado, 2016].
3. La solución para facilitar la interoperabilidad y la mantenibilidad de los procesos, garantizando la independencia del proceso modelado frente al lenguaje de modelado seleccionado [García-Borgoñón, 2016].

9.1.2. Línea de investigación en ITAINNOVA

Este trabajo de tesis se ha desarrollado en colaboración con ITAINNOVA, el Instituto Tecnológico de Aragón, un Centro Tecnológico de carácter público dependiente del Departamento de Innovación, Investigación y Universidad del Gobierno de Aragón. ITAINNOVA dispone de un plan científico tecnológico con el objetivo de generar y adquirir conocimiento científico y tecnológico por parte del personal técnico, con la finalidad de obtener resultados que impacten lo máximo posible en los productos y servicios de las empresas de Aragón, España y Europa.

Dentro de ese plan la línea de investigación de ingeniería del software dirigida por modelos se centra en la aplicación práctico del enfoque MDE a nivel de proceso (Model-Based Process Engineering) y de producto (Model-based Software Engineering). En particular, se pretende investigar en la creación de lenguajes y modelos que permitan describir y ejecutar procesos de negocio, para abordar los siguientes retos:

- La definición y ejecución de procesos de negocio, en particular en los dominios software y logístico.
- La toma de decisiones colaborativa (en su aplicación al desarrollo y mantenimiento de sistemas y a la gestión de la cadena de suministro)
- El desarrollo de sistemas forma distribuida y en colaboración (Global Software Engineering)

Además de las aportaciones que este trabajo realiza sobre la línea de investigación de MDE de ITAINNOVA, la componente colaborativa e interorganizacional ha venido influenciada por la pertenencia al Centro Nacional de Conocimiento en la aplicación de las Tecnologías de la Información y de las Comunicaciones a la resolución de problemas en el ámbito de la logística. Este grupo surge en 2008 como una propuesta de enfocar la creación de sistemas basados en las nuevas Tecnologías de la Información y de las Comunicaciones para resolver problemas en el ámbito del transporte y la logística. La propuesta que fue reconocida por el Ministerio de Industria, dentro del Plan Avanza, como Centro Nacional de Conocimiento, con el objetivo de aplicar las TICs para desarrollar una logística colaborativa y sostenible (REF TSI-070200-2008-76).

9.1.3. Relaciones con otros organismos que han influido en el desarrollo de este trabajo de tesis

A continuación se nombran algunos de los grupos de investigación, organismos y empresas con los que el grupo IWT2 se encuentra actualmente trabajando entorno a esta línea de procesos de negocio, y que son relevantes tanto en el desarrollo de este trabajo de tesis como a la hora de abordar nuevas líneas de trabajo futuro.

- **El Grupo de Innovación Tecnológica de la fundación FISEVI.** El grupo IWT2 tiene una estrecha colaboración con el Grupo de Innovación Tecnológica (GiT), un grupo de investigación relacionado con la I+D+i y las nuevas tecnologías que se encuentra bajo la coordinación del gerente de la Fundación Pública Andaluza para la Gestión de la Investigación en Salud de Sevilla (FISEVI)¹. Fruto de esta colaboración se está planteando la aplicación de los resultados de esta tesis en el ámbito de los procesos clínicos y hospitalarios, dado que junto a la logística, como vimos en los capítulos iniciales, son los dominios de mayor aplicación de los procesos intensivos en conocimiento.
- **El Grupo de Investigación en Ingeniería del Software de la Universidad de Oviedo.** Aunque algo más adelante será destacado como una futura línea de trabajo común entre ambos grupos de investigación, ya se han realizado las primeras propuestas para incorporar técnicas de pruebas de software con el objetivo de asegurar la conformidad de los procesos ejecutados conforme a su definición. Esto es posible gracias a que el Grupo de Investigación de Ingeniería del Software (GIIS) de la Universidad de Oviedo tiene como línea de trabajo la definición de metodologías y herramientas para la calidad del software, especialmente en el ámbito de las pruebas de software.
- **El Grupo de Investigación en Mejora del Proceso Software y Métodos Formales de la Universidad de Cádiz.** Como veremos en los trabajos futuros una de las aplicaciones más directas del modelado de procesos colaborativos es la simulación de los mismos, como una técnica de apoyo a la toma de decisiones antes de llevarlas a cabo en el dominio de la CdS. Con el grupo de Mejora del Proceso Software y Métodos Formales (SPI&FM) de la Universidad de Cádiz se ha comenzado a trabajar y se profundizará en la colaboración para derivar el modelo de forma automática hacia su simulación.
- **El Laboratorio de Investigación y Formación en Informática Avanzada de la Universidad Nacional de La Plata, Argentina.** Conjuntamente con el Laboratorio de Investigación y Formación en Informática Avanzada (LIFIA) de la Universidad Nacional de La Plata (Argentina) se vienen realizando desde hace tiempo trabajos de investigación para mejorar los procesos de la metodología NDT, concretamente en el proceso de ingeniería de requisitos en lo que tiene que ver con la conciliación de los requerimientos. Estos trabajos conjuntos han propiciado el establecimiento de convenios de colaboración para la realización de estancias y proyectos internacionales, así como tesis doctorales conjuntas.
- **Southampton Solent University.** En los últimos años se viene trabajando en estrecha colaboración con la Doctora emérita Margaret Ross en el ámbito de la calidad del software, lo que ha propiciado la realización de estancias en dicha universidad y la revisión por su parte tanto de los trabajos de investigación, y así como de los artículos desarrollados.

¹La Fundación FISEVI surge con el objetivo de dar servicio a los investigadores de los centros del Sistema Sanitario Público de Andalucía en la provincia de Sevilla, aglutinando las actividades que anteriormente eran desarrolladas por otras entidades de gestión. Su cartera de servicios se compone de todo un conjunto de actividades de apoyo al desarrollo de la investigación en Salud en el seno del Sistema Sanitario Público de Andalucía, así como la gestión de ayudas y contratos que permiten las actividades de investigación de los diferentes grupos de los centros de dicho Sistema Sanitario ubicados en la provincia de Sevilla.

Todos estos son ejemplos de colaboración con otros grupos de investigación, organismos, entidades públicas y empresas en los que la línea estratégica de procesos de negocio del grupo IWT2 tiene especial protagonismo y con los que, de alguna u otra forma, se ha trabajado en el desarrollo de esta tesis, tal y como queda de manifiesto en el anexo D donde se recopilan los principales trabajos, artículos, proyectos y estancias del doctorando.

El contexto dentro del grupo IWT2 y las relaciones con otros grupos han conformado el marco estratégico en el que se ha desarrollado este trabajo de tesis, que ha constituido una importante influencia a la hora de afrontar las soluciones planteadas y la manera en la que se han llevado a cabo, de ahí el hecho de que lo hayamos destacado como un aspecto relevante. Una vez precisado este marco contextual, en la siguiente sección se exponen las principales aportaciones obtenidas en el desarrollo de este trabajo de tesis.

9.2. Aportaciones de esta tesis

Una vez descrito y documentado el trabajo realizado durante la realización de esta tesis doctoral y expuesto el marco de trabajo en el que se ha llevado a cabo, es momento para recapitular las principales aportaciones obtenidas como resultado de la misma. En cada una de estas aportaciones haremos referencia al objetivo que al comienzo del trabajo se planteó para observar que existe al menos una correspondencia para todos los objetivos, con lo que se puede concluir que todos ellos han sido cubiertos.

9.2.1. Un estudio del estado del arte sobre lenguajes de modelado de procesos de la CdS

El primer paso que nos planteamos fue el conocer el estado de la situación actual en relación a las diversas iniciativas existentes para modelar procesos en la CdS. Este análisis nos permitió conocer con mayor grado de detalle las propuestas ya realizadas, los problemas que han intentado abordar así como las líneas de trabajo abiertas en la temática, de manera que tras este estudio pudimos establecer de forma más concreta los objetivos específicos a abordar.

Para llevar a cabo este estudio apostamos por la guía propuesta por Kitchenham y conocida como revisión sistemática de la literatura (Systematic Literature Review, SLR), dado que es la más ampliamente aceptada en el campo de la ingeniería del software [Kitchenham y Charters, 2007].

A la vista de los resultados evidenciamos una evolución del nivel de abstracción de los modelos de procesos de la CdS, donde lo relevante empieza a ser entender y analizar las relaciones entre los participantes, para lo cual el paradigma basado en modelos está siendo el más utilizado. Como principales conclusiones de la revisión obtuvimos la necesidad de disponer de una visión holística en la que las decisiones se tomen conjuntamente, por lo que se hacía preciso:

- Establecer las bases teóricas que permitan obtener una definición comúnmente aceptada del problema.
- Desarrollar estudios empíricos para analizar cuantitativamente barreras y mejoras.
- Crear un enfoque multi-disciplinar combinando resultados de otras áreas de investigación.

Por todo ello esta primera contribución puede ser la base de otros investigadores en el ámbito del modelado de procesos o casos de negocio, y nos sirvió como punto de partida para evidenciar la necesidad de poder abordar el modelado de procesos de negocio de la CdS con un nuevo enfoque, bottom-up, en el que se dé soporte a una diversidad de lenguajes y herramientas y se reutilicen los modelos existentes.

9.2.2. Un lenguaje de modelado de procesos de negocio colaborativos

Nos proponíamos como objetivo modelar procesos negocios colaborativos conforme a un lenguaje que nos diera dos características:

- Permitiera reutilizar modelos existentes, siguiendo un enfoque bottom-up a partir de los los modelos de procesos individuales de cada organización.
- Se centrara en las relaciones entre los procesos garantizando la privacidad de la visión interna de cada participante así como su autonomía a la hora de tomar decisiones.

Para poder satisfacer la primera condición hemos tomado como base INROMA, un lenguaje auxiliar para representar los conceptos fundamentales que describen un proceso. El hecho de que INROMA disponga de un método para, a partir de un lenguaje de modelado de procesos, generar de forma automática el modelo INROMA equivalente, hace que ese primer paso bottom-up desde múltiples lenguajes de modelado de procesos quede cubierto, motivo que justifica la elección de esta propuesta como base.

Sin embargo, su versión actual no da soporte para expresar los puntos relativos a la colaboración entre los mismos, como la vista multi organizacional y el modelado de actividades y productos en colaboración o la forma de analizar mediante métricas e indicadores dicho proceso colaborativo. Por ello, para cumplir la segunda condición nuestra propuesta, que hemos denominado CBGProcess, extiende INROMA para poder dar soporte al modelado de procesos colaborativos.

9.2.3. Un método para construir el modelo de proceso colaborativo

La tercera contribución se basa en un método para, a partir de un conjunto de procesos individuales descritos con INROMA, generar un proceso colaborativo utilizando el nuevo meta-modelo CBGProcess.

Este método permite separar la vista individual de un proceso de una organización de la vista colaborativa, manteniendo por tanto la privacidad y la autonomía en la toma de decisiones y compartiendo aquellos elementos que cada organización quiera para poder conformar el proceso en colaboración. Dicho método ha quedado enunciado con sus precondiciones, acciones y postcondiciones, y ha sido utilizado en la validación tanto teórica como mediante casos reales.

Adicionalmente este método se complementa con una transformación M2M horizontal entre INROMA y su extensión CBGProcess para generar la vista colaborativa desde un proceso individual. Hemos detallado todas las relaciones de la transformación utilizando QVT Relations de forma gráfica.

9.2.4. Un lenguaje de modelado de casos colaborativos

Para dar soporte a la adaptación y flexibilidad de los procesos colaborativos, dentro de nuestra propuesta se hace preciso que los procesos cuenten con una parte estática y una dinámica, relativa al tiempo de ejecución.

Durante el desarrollo de este trabajo hemos detallado cómo el paradigma de gestión de casos es idóneo para, de forma complementaria a la gestión de procesos, poder aportar las características de dinamismo, flexibilidad y adaptación presentes en los modelos de procesos en colaboración. En este punto hemos justificado tanto el uso del paradigma de gestión de casos para entornos colaborativos en la CdS como, a través de sus objetivos de diseño y del estado del arte en la temática, motivar la necesidad de plantear un nuevo metamodelo y no adoptar CMMN.

Como resultado hemos propuesto CBGCase, un lenguaje para representar casos en colaboración que incorpora la capacidad de planificación dinámica sobre la estructura estática inicialmente planteada, manteniendo los aspectos relativos a la colaboración ya presentes en el dominio de procesos.

9.2.5. Una transformación para convertir los modelos de procesos de negocio colaborativos en modelos de casos colaborativos

Nos propusimos definir un marco conceptual basado en transformaciones entre modelos que nos permitiera convertir los modelos de procesos de negocio colaborativos a modelos de casos colaborativos, añadiendo toda la capacidad de planificación dinámica de los últimos. De esta forma seríamos capaces de mantener un enfoque de reutilización de modelos incluso a partir del modelo de proceso colaborativo.

Por ello hemos definido:

- Un método para, a partir de un modelo de proceso colaborativo descrito con CBGProcess, generar un modelo de caso colaborativo conforme a CBGCase que nos permite incluir la capacidad de planificación dinámica, mantenimiento la trazabilidad con el dominio de procesos colaborativos y a su vez con las vistas individuales que cada organización tiene de la colaboración.
- Una transformación M2M descrita con QVT Relations de forma gráfica que permite, a partir del metamodelo CBGProcess extendido desde INROMA, generar automáticamente una base de modelo de casos colaborativos conforme a CBGCase.

9.2.6. Una herramienta de soporte para su aplicación práctica

Para que este marco conceptual que pueda ser utilizado en la práctica por organizaciones nos propusimos desarrollar una herramienta de soporte que hemos denominado CBG-Tool, y que ofrece:

- La capacidad de editar modelos de procesos conformes a CMGProcess, a través de la implementación de un perfil UML y su despliegue en el contexto de ejecución de Enterprise Architect.
- La implementación de la transformación M2M horizontal de INROMA a CBGProcess, mediante la programación de las relaciones dentro de un Add-in.
- La capacidad de editar modelos de procesos conformes a CMGCase, a través de la implementación de un perfil UML y su despliegue en el contexto de ejecución de Enterprise Architect.
- La implementación de la transformación M2M de CBGProcess a CBGCase, mediante la programación de las relaciones dentro de un Add-in.

Hemos validado estas funcionalidades atendiendo a escenarios ficticios que representan las situaciones reales que han justificado la realización de este trabajo de tesis y que fueron descritas en los capítulos iniciales, haciendo uso paso a paso de la solución CBG-Tool a nivel práctico.

9.2.7. Validación de la propuesta con casos reales

Una vez desarrollados el marco CBG y la herramienta de soporte CBG-Tool, el último objetivo a cumplir era validar la propuesta con casos reales, como una forma de corroborar las hipótesis que se habían planteado a la hora de formular el problema y poder observar que la solución propuesta cumplía con las expectativas planteadas.

Para ello hemos utilizado el caso de estudio real del proceso de compra y distribución de pescado fresco, desde las lonjas hasta el punto de venta, pasando por unas plataformas logísticas de manipulación y almacenaje del producto dispersas por todo el territorio nacional, de la organización CARREFOUR en España.

Siguiendo los métodos definidos y haciendo uso de la solución CBG-Tool, hemos ido generando la vista individual de cada proceso, su visión particular dentro de la colaboración así como el modelo de proceso colaborativo. En este sentido hemos podido validar que nuestro enfoque permite:

- Generar la vista colaborativa a partir de los modelos de procesos individuales.
- Disponer de una vista individual y particular de cada organización dentro de la colaboración, manteniendo su privacidad en el proceso y su autonomía a la hora de tomar decisiones.
- Tener una trazabilidad entre las actividades y productos en colaboración con la vista individual de cada organización, de forma que se pueda dotar de flexibilidad a la misma en cuanto a su posible participación en varias CdS.

Por último, continuando con el método propuesto y haciendo uso de las transformaciones de procesos a casos dentro de la herramienta CBG-Tool, hemos validado cómo se podría incluir la capacidad de dinamismo para replanificar en tiempo de ejecución las posibles tareas, generando un modelo de caso colaborativo. En este aspecto hemos podido validar que nuestro enfoque permite:

- Incorporar dinamismo a partir del modelo de proceso colaborativo, generando de forma automática su equivalente en el dominio de casos.
- Mantener la trazabilidad desde el modelo de caso hacia atrás, de forma que podamos en tiempo de ejecución replanificar tareas del proceso colaborativo y, a su vez, cada organización de la CdS pueda identificar aquellas tareas que sólo forman parte de su vista individual de la colaboración.

De forma general, esta validación con un proceso complejo real nos ha permitido evidenciar que el marco CBG propuesto en este trabajo de tesis permite obtener una vista fiable de cómo se lleva a cabo un proceso en colaboración, en menos tiempo, con menos errores y facilitando la mantenibilidad de los modelos generados al poder incluir las características de flexibilidad y dinamismo requeridas en el dominio de la CdS.

La utilización de casos reales derivados de proyectos con empresas para realizar esta validación, no sólo ha servido para alcanzar dicho objetivo sino que se han abierto nuevas ideas y líneas de trabajo entorno a esta temática y con una importante carga de investigación aplicada a la industria, con la continuidad que esto supone en la temática de los procesos.

Todas y cada una de estas aportaciones nos han permitido alcanzar los objetivos planteados al comienzo del trabajo de tesis, reforzando las hipótesis y evidenciado las principales fortalezas de la solución propuesta.

9.2.8. Aportaciones no tecnológicas

Además de todas las aportaciones en el ámbito científico-tecnológico anteriormente presentadas, este trabajo de tesis ha aportado una nueva forma de abordar el modelado de procesos de negocio colaborativos. Estas ideas podrán ser desarrolladas por la comunidad con nuevas técnicas, pero suponen en sí mismas unas aportaciones novedosas a nivel de ideas y conceptos que resumimos a continuación:

- Un enfoque *bottom-up*: la forma de abordar el modelado partiendo y reutilizando los modelos de procesos individuales que cada organización tuviera es una idea novedosa que permite: 1) poner en primer lugar a cada organización, por encima de las colaboraciones en las que participe; 2) reduce el riesgo de tener incoherencias entre el modelo individual y el colaborativo y; 3) da soporte a que cada compañía utilice el lenguaje de modelado de procesos de negocio más idóneo para sus necesidades.
- Nuestra solución permite que una misma colaboración pueda ser vista de manera diferente por diversos miembros de la CdS. Hasta ahora, los enfoques *top-down* establecían el modelo de colaboración que determinaba si estábamos ante un proceso estructurado, un caso, un proceso dirigido por eventos o datos, etc. Todos los miembros de la CdS debían de ver el escenario de la misma manera. Con el marco CBG, cada compañía mantiene su autonomía no sólo en la forma en la que modela sus procesos individuales, sino en cómo percibe cada una de las colaboraciones. Lo que para una organización puede ser un proceso completamente estructurado, para otra puede ser visto, modelado y gestionado como un caso o como un proceso intensivo en conocimiento.
- La autonomía en la toma de decisiones y la privacidad en lo relativo al proceso interno de una compañía, si bien se podían acotar con los enfoques existentes, debían de ser establecidos de forma general para cualquier colaboración, o en su defecto hacer adaptaciones específicas ad-hoc no soportadas en los lenguajes de modelado. La solución CBG se concibe, partiendo del proceso individual, para cada colaboración, por lo que es posible que en una CdS una organización quiera mantener mucha autonomía y privacidad y en otra lo contrario, dado que los lenguajes de modelado parten de la vista individual y sobre esa para cada CdS se establecen los elementos visibles a la colaboración.
- Las propuestas basadas en enfoques *top-down* parten de una estructura establecida de una colaboración y, aunque los cambios y modificaciones se pueden llevar a cabo, son más viables en CdS que requieren poco dinamismo. Nuestro enfoque está orientado a CdS muy dinámicas, en las que los cambios son no sólo posibles sino bienvenidos, como método para dar respuesta a las condiciones del mercado en el que operan las empresas en la actualidad. A través de la vista individual, se pueden añadir fácilmente las nuevas coreografías requeridas para colaborar en una nueva CdS o para colaborar con una nueva organización en la cadena.

9.3. Trabajos futuros

Una vez recapituladas las principales aportaciones realizadas con el desarrollo de este trabajo de tesis, es importante resaltar que es un camino que no termina, sino que sus resultados, en confluencia con los otros trabajos del grupo IWT2 que hemos comentado anteriormente, han propiciado la apertura de nuevas ideas y líneas para avanzar en esta temática. A continuación se plantean algunos de los trabajos futuros más representativos en el ámbito de los procesos.

9.3.1. La capacidad de simular modelos de procesos y casos colaborativos

Una de las aplicaciones más comunes cuando se generan modelos formales de procesos es poder analizar el efecto de realizar cambios antes de producirlos de forma real en el proceso. Para ello se hace uso de las técnicas de simulación. La simulación nos permite monitorizar los procesos, analizarlos, realizar cambios, y nos ayuda a tomar la decisión óptima sin necesidad de experimentar en un entorno real [2010].

Para ello se crea un modelo de simulación, que refleja la realidad del proceso y que es procesado en un entorno informático [Van der Aalst *et al.*, 2010]. Los modelos de simulación se usan típicamente cuando se quiere analizar las consecuencias sobre una parte muy concreta del proceso, por lo que es posible modelar de forma detallada los elementos de esa parte y obtener conclusiones fiables. Alternativamente esta técnica puede ser utilizada en el caso de que realizar un experimento en real sea muy costoso o incluso imposible. La simulación nos ofrece una alternativa para analizar datos y apoyar la toma de decisiones sin tener que establecer un entorno real en el que se produzcan los cambios.

A partir de nuestro trabajo una clara línea de futuro es generar de forma automática modelos de simulación de los procesos y casos colaborativos, atendiendo por su naturaleza a las técnicas de eventos discretos (Discrete Event Simulation, DES) y basada en agentes (Agent Based Modeling and Simulation, ABMS). A partir de los modelos de simulación, podremos realizar diversos análisis:

- Análisis de situación (As Is - To Be) para identificar el comportamiento esperado del real.
- Análisis de sensibilidad de cómo afecta un cambio en el proceso.
- De forma especial, el análisis de cómo el proceso colaborativo puede mejorarse en una CdS [Wang *et al.*, 2011].

9.3.2. La capacidad de ejecutar procesos y casos colaborativos

La segunda aplicación más extendida a partir del modelado de los procesos de negocio tiene que ver con dar capacidad de ejecución, como un mecanismo para facilitar que en el día a día las actividades desarrolladas son fieles al comportamiento esperado y representado en el modelo de proceso.

Para ello el método más sencillo es realizar transformaciones M2T desde los modelos CBG-Process y CBGCase a los lenguajes XML que algunos de los motores open source de ejecución de procesos y casos más utilizados, como por ejemplo Activity ² o Camunda ³.

Otra opción sería llevar a cabo transformaciones M2M desde los metamodelos CBGProcess y CBGCase a BPMN y CMMN, para a partir de ahí poder hacer uso de los motores de ejecución, si bien en este caso hemos detectado el riesgo de que por lo general cada motor de ejecución soporta una parte de los metamodelos BPMN y CMMN, así como que la forma de describir el modelo suele ser en un XML no estandarizado.

9.3.3. La capacidad de orientar el modelo a procesos intensivos en conocimiento

Como vimos en las influencias tecnológicas en la CdS los procesos que se desarrollan entre las entidades se rigen por lo que se conoce en la literatura como procesos intensivos en conocimiento (KIPs), en los que cobra especial importancia el papel de las personas que desarrollan el proceso y que toman decisiones atendiendo a la información del entorno y su experiencia[Hauder *et al.*, 2014].

En este sentido una posible línea de futuro tiene que ver con la inclusión en el modelo de los criterios en los que se basan los actores del proceso a la hora de tomar las decisiones y las consecuencias que tienen las mismas, con el objetivo de crear una base de conocimiento que retroalimente el proceso y se incorpore el conocimiento de las personas al propio modelo de proceso.

El resultado de esta actividad sería un modelo de proceso en el que las decisiones y sus consecuencias están caracterizadas y analizadas sistemáticamente, de manera que nuevos actores que entran en el proceso que requiere de conocimiento, reducen su curva de aprendizaje al incorporar la experiencia previa acumulada en la organización o en la CdS colaborativa.

²<http://activiti.org>

³<http://camunda.org>

9.3.4. La capacidad de incorporar el factor humano en el modelo

Asociada a la línea anterior en procesos intensivos en conocimiento, tras la revisión de la literatura evidenciamos que existe un tema que todavía no ha sido tratado: el modelado del factor humano[de Man, 2009]. Este punto hace referencia a incorporar en el modelo el comportamiento que tienen los trabajadores de un proceso o caso y cómo afecta el grado de libertad de las actividades a incluir o a seleccionar en el mismo.

Esta línea se podría orientar a resolver estas cuestiones: ¿Cómo afecta en el proceso que un determinado puesto de decisión sea cubierto por una persona con más o menos experiencia?. ¿Qué puntos del proceso son más sensibles a las personas y su conocimiento?. Y de manera formal obtener las evidencias como una herramienta de apoyo a la toma de decisiones para cubrir estos puestos con los perfiles pertinentes.

En un enfoque colaborativo, esta misma línea es extensible al ámbito de una organización, de forma que podamos caracterizar cómo influye una organización en otras y poder seleccionar de forma dinámica aquellos colaboradores idóneos para formar una cadena de valor.

9.3.5. La definición de un mecanismo que facilite las auditorías y certificaciones de calidad de los procesos

Habitualmente las organizaciones, una vez definidos e institucionalizados los procesos, suelen querer acceder a las certificaciones que acreditan la obtención de alguna norma o estándar. Sin embargo, las tareas relacionadas específicamente a la preparación y ejecución de estas auditorías es un coste bastante elevado para cualquier organización, puesto que los objetivos de estas tareas están dirigidos a recopilar las evidencias necesarias para poder pasar con éxito las evaluaciones de tales certificaciones.

Una línea de trabajo en este sentido consistiría en incorporar en el modelo de proceso los aspectos susceptibles de ser tomados como evidencias a la hora de realizar las certificaciones, estableciendo un marco de evaluación en el que se pudieran incorporar diferentes normas y estándares de todo tipo.

En la CdS colaborativa el enfoque podría no estar orientado a la certificación en su totalidad, aunque la integración con los sistemas informáticos de las organizaciones para recoger las evidencias al menos daría la evidencia de que el proceso se cumple en la realidad.

9.4. Conclusiones

A lo largo de toda esta memoria hemos resumido los resultados obtenidos en cada una de las tareas llevadas a cabo dentro del este trabajo de tesis, desde la puesta en contexto y la definición de las necesidades detectadas tras la revisión sistemática de literatura, hasta la propuesta de solución, inicialmente desde un punto de vista más teórico, para llegar a un enfoque totalmente práctico con la herramienta de soporte y la manifestación de su utilidad mediante la exposición de los casos de estudio.

En este capítulo hemos presentado los principales resultados que hemos obtenido, notablemente influenciados por el contexto en el que se ha desarrollado la actividad, tanto el grupo IWT2 como sus relaciones con otros grupos de investigación y empresas con los que se trabaja en el ámbito de los procesos de negocio, conformando todo ello un marco estratégico para esta línea de investigación.

También hemos detallado las principales aportaciones de la presente tesis doctoral, alineadas a los objetivos inicialmente planteados, así como las nuevas líneas de trabajo que se abren dentro de este marco estratégico y que a buen seguro complementarán los resultados obtenidos.

Bibliografía

- [Aguilar-Saven, 2004] Aguilar-Saven, R. S. (2004). Business process modelling: Review and framework. *International Journal of production economics*, 90(2):129–149.
- [Ahuja y Carley, 1998] Ahuja, M. K. y Carley, K. M. (1998). Network structure in virtual organizations. *Journal of Computer-Mediated Communication*, 3(4):0–0.
- [Ajmera y Cook, 2009] Ajmera, A. y Cook, J. (2009). A multi-phase framework for supply chain integration. *SAM Advanced Management Journal*, 74(1):37.
- [Aleem *et al.*, 2012] Aleem, S., Lazarova-Molnar, S., y Mohamed, N. (2012). Collaborative business process modeling approaches: A review. En *Proc. of the 2012 IEEE 21st International workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises*, pp. 274–279.
- [Alver, 2012] Alver, M. O. (2012). JabRef reference Manager. <http://jabref.sourceforge.net/>. [Online; accessed 13-January-2014].
- [Angerhofer y Angelides, 2006] Angerhofer, B. J. y Angelides, M. C. (2006). A model and a performance measurement system for collaborative supply chains. *Decision Support Systems*, 42(1):283–301.
- [APICS Supply Chain Council, 2013] APICS Supply Chain Council (2013). APICS Supply Chain Council Dictionary. [Online; accessed 29-Oct-2014].
- [Arévalo Maldonado, 2016] Arévalo Maldonado, C. (2016). Una propuesta basada en el paradigma dirigido por modelos para la extracción de procesos del software desde sistemas heredados utilizando la perspectiva temporal.
- [Arshinder *et al.*, 2011] Arshinder, K., Kanda, A., y Deshmukh, S. (2011). A review on supply chain coordination: Coordination mechanisms, managing uncertainty and research directions.
- [Ashton y Kelly, 2006] Ashton, H. y Kelly, D. (2006). The business impact of bpm with soa: Building a business case for bpm with soa roi. *Upside Research, Inc. www.upsideresearch.com*.
- [Attaran y Attaran, 2007] Attaran, M. y Attaran, S. (2007). Collaborative supply chain management: the most promising practice for building efficient and sustainable supply chains. *Business Process Management Journal*, 13(3):390–404.
- [Azaiez, 2007] Azaiez, S. (2007). *Model driven development approach for multiagent systems*. Theses, Université de Savoie.

- [Back y Wright, 2012] Back, R.-J. y Wright, J. (2012). *Refinement calculus: a systematic introduction*. Springer Science & Business Media.
- [Badole *et al.*, 2012] Badole, C. M., Jain, D. R., Rathore, D. A., y Nepal, D. B. (2012). Research and opportunities in supply chain modeling: A review.
- [Bajwa *et al.*, 2008] Bajwa, I. S., Kazmi, R., Mumtaz, S., Choudhary, M. A., y Naweed, M. S. (2008). Soa and bpm partnership: A paradigm for dynamic and flexible process and it management. *World Academy of Science, Engineering and Technology*, 45(4):16–22.
- [Bajwa *et al.*, 2009] Bajwa, I. S., Samad, A., Mumtaz, S., Kazmi, R., y Choudhary, A. (2009). Bpm meeting with soa: a customized solution for small business enterprises. En *Information Management and Engineering, 2009. ICIME'09. International Conference on*, pp. 677–682. IEEE.
- [Baker y Byler, 1983] Baker, J. C. y Byler, E. U. (1983). Swift: a fast method to facilitate international financial transactions. *Journal of World Trade*, 17(5):458–465.
- [Basak, 2015] Basak, S. K. (2015). A comparison of three reference management software: Jabref, zotero, and endnote. *International Journal of Research in Information Technology*, 3(4):223–231.
- [Bechini *et al.*, 2008] Bechini, A., Cimino, M. G., Marcelloni, F., y Tomasi, A. (2008). Patterns and technologies for enabling supply chain traceability through collaborative e-business. *Information and Software Technology*, 50(4):342–359.
- [Berre, 2008] Berre, A. (2008). Service oriented architecture modeling language (soaml)-specification for the uml profile and metamodel for services (upms). *Object Management Group (OMG)*.
- [Bézivin, 2005] Bézivin, J. (2005). On the unification power of models. *Software & Systems Modeling*, 4(2):171–188.
- [Bhattacharya *et al.*, 2007] Bhattacharya, K., Caswell, N. S., Kumaran, S., Nigam, A., y Wu, F. Y. (2007). Artifact-centered operational modeling: Lessons from customer engagements. *IBM Systems Journal*, 46(4):703–721.
- [Bhombal,] Bhombal, F. Soa and bpm: Service oriented architecture and business process management–primac systems road map for 2007. *PRIMAC Systems Inc. fbhombal@vercom.com*.
- [Bouchbout y Alimazighi, 2011] Bouchbout, K. y Alimazighi, Z. (2011). Inter-organizational business processes modelling framework. En *ADBIS (2)*, pp. 45–54. Citeseer.
- [Bowersox, 1990] Bowersox, D. J. (1990). *The strategic benefits of logistics alliances*. Harvard Business Review.
- [Brambilla *et al.*, 2012a] Brambilla, M., Cabot, J., y Wimmer, M. (2012a). *Model-Driven Software Engineering in Practice*. Morgan & Claypool Publishers, 1st edición.
- [Brambilla *et al.*, 2012b] Brambilla, M., Cabot, J., y Wimmer, M. (2012b). *Model-driven software engineering in practice*, volumen 1. Morgan & Claypool Publishers.

- [Breitenmoser y Keller, 2015] Breitenmoser, R. y Keller, T. (2015). Case management model and notation-a showcase. *European Scientific Journal*, 11(25).
- [Camarinha-Matos y Afsarmanesh, 2008] Camarinha-Matos, L. M. y Afsarmanesh, H. (2008). On reference models for collaborative networked organizations. *International Journal of Production Research*, 46(9):2453–2469.
- [Cassivi, 2006] Cassivi, L. (2006). Collaboration planning in a supply chain. *Supply Chain Management: An International Journal*, 11(3):249–258.
- [Cetinkaya y Verbraeck, 2011] Cetinkaya, D. y Verbraeck, A. (2011). Metamodeling and model transformations in modeling and simulation.
- [Chandra, 2007] Chandra, C. (2007). *Ontology as information system support for supply chain management*. cited By 0.
- [Chandra, 2008] Chandra, C. (2008). *Supply chain workflow modeling using ontologies*. cited By 2.
- [Chao *et al.*, 2009] Chao, T., Cohn, D., Flatgard, A., Hahn, S., Linehan, M., Nandi, P., Nigam, A., Pinel, F., Vergo, J., y y Wu, F. (2009). Artifact-based transformation of ibm global financing. En *Business Process Management*, pp. 261–277. Springer.
- [Chatfield *et al.*, 2009] Chatfield, D. C., Harrison, T. P., y Hayya, J. C. (2009). Scml: An information framework to support supply chain modeling.
- [Choi *et al.*, 2008] Choi, Y., Kang, D., Chae, H., y Kim, K. (2008). An enterprise architecture framework for collaboration of virtual enterprise chains.
- [Chopra y Meindl, 2007] Chopra, S. y Meindl, P. (2007). *Supply chain management. Strategy, planning & operation*. Springer.
- [Cognini *et al.*, 2016] Cognini, R., Corradini, F., Gnesi, S., Polini, A., y Re, B. (2016). Business process flexibility-a systematic literature review with a software systems perspective. *Information Systems Frontiers*, pp. 1–29.
- [Cooper *et al.*, 1997] Cooper, M. C., Lambert, D. M., y Pagh, J. D. (1997). Supply chain management: more than a new name for logistics.
- [Council of Supply Chain Management Professionals, 2013] Council of Supply Chain Management Professionals (2013). Supply Chain Management Terms and Glossary. [Online; accessed 29-Oct-2014].
- [Cousins y Menguc, 2006] Cousins, P. D. y Menguc, B. (2006). The implications of socialization and integration in supply chain management. *Journal of Operations Management*, 24(5):604–620.
- [CUI y Liu, 2005] CUI, Z.-d. y Liu, J. (2005). Supply chain modeling and analysis based on generalized stochastic petri nets. *Systems Engineering-theory & Practice*, 12:002.
- [Czarnecki y Helsen, 2006] Czarnecki, K. y Helsen, S. (2006). Feature-based survey of model transformation approaches. *IBM Systems Journal*, 45(3):621–645.

- [Da Silva *et al.*, 2011] Da Silva, F. Q., Santos, A. L., Soares, S., França, A. C. C., Monteiro, C. V., y Maciel, F. F. (2011). Six years of systematic literature reviews in software engineering: An updated tertiary study. *Information and Software Technology*, 53(9):899–913.
- [Daaboul *et al.*, 2012] Daaboul, J., Bernard, A., y Laroche, F. (2012). Extended value network modelling and simulation for mass customization implementation. *Journal of Intelligent Manufacturing*, 23(6):2427–2439.
- [Daaboul *et al.*, 2014] Daaboul, J., Castagna, P., Da Cunha, C., y Bernard, A. (2014). Value network modelling and simulation for strategic analysis: A discrete event simulation approach. *International Journal of Production Research*, 52(17):5002–5020. cited By 1.
- [Damaggio *et al.*, 2013] Damaggio, E., Hull, R., y Vaculín, R. (2013). On the equivalence of incremental and fixpoint semantics for business artifacts with guard–stage–milestone lifecycles. *Information Systems*, 38(4):561–584.
- [Damelio, 2011] Damelio, R. (2011). *The basics of process mapping*. CRC Press.
- [Damodaran, 2004] Damodaran, S. (2004). B2b integration over the internet with xml: Rosettanet successes and challenges. En *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, pp. 188–195. ACM.
- [Davenport, 2013a] Davenport, T. H. (2013a). *Process innovation: reengineering work through information technology*. Harvard Business Press.
- [Davenport, 2013b] Davenport, T. H. (2013b). *Thinking for a living: how to get better performances and results from knowledge workers*. Harvard Business Press.
- [Davenport y Beers, 1995] Davenport, T. H. y Beers, M. C. (1995). Managing information about processes. *Journal of Management Information Systems*, pp. 57–80.
- [Davenport *et al.*, 1988] Davenport, T. H., Hammer, H., y Metsisto, T. J. (1988). How executives can shape their company's information systems. *Harvard Business Review*, 67(2):130–134.
- [Davenport y Short, 1990] Davenport, T. H. y Short, J. E. (1990). The new industrial engineering: information technology and business process redesign. *Sloan management review*, 31(4).
- [Davulcu *et al.*, 1999] Davulcu, H., Kifer, M., Pokorny, L. R., Ramakrishnan, C., Ramakrishnan, I., y Dawson, S. (1999). Modeling and analysis of interactions in virtual enterprises. En *Research Issues on Data Engineering: Information Technology for Virtual Enterprises, 1999. RIDE-VE'99. Proceedings., Ninth International Workshop on*, pp. 12–18. IEEE.
- [Dayal *et al.*, 2001] Dayal, U., Hsu, M., y Ladin, R. (2001). Business process coordination: State of the art, trends, and open issues. En *VLDB*, volumen 1, pp. 3–13.
- [de Man, 2009] de Man, H. (2009). Case management: A review of modeling approaches. *BPTrends, January*, 2009.
- [de Oliveira *et al.*, 2010] de Oliveira, L. B. R., Felizardo, K. R., Feitosa, D., y Nakagawa, E. Y. (2010). Reference models and reference architectures based on service-oriented architecture: a systematic review. En *Software Architecture*, pp. 360–367. Springer.

- [Del Fabro *et al.*, 2005] Del Fabro, Marcos Didonet and Bézivin, Jean and Jouault, Frédéric and Breton, Erwann and Gueltas, Guillaume and others (2005). AMW: a generic model weaver. *Journées sur l'Ingénierie Dirigée par les Modèles*, pp. 105–114.
- [Di Ruscio *et al.*, 2012] Di Ruscio, D., Eramo, R., y Pierantonio, A. (2012). Model transformations. En *Formal Methods for Model-Driven Engineering*, pp. 91–136. Springer.
- [Dietrich J, 2006] Dietrich J, Hofreiter B, H. C. L. P. S. R. Z. M. (2006). Un/cefact modelling methodology (umm), umm foundation module v1.0. Technical report, UN/CEFACT.
- [Dogac *et al.*, 2004] Dogac, A., Kabak, Y., Laleci, G., Sinir, S., Yildiz, A., Kirbas, S., y Gurcan, Y. (2004). Semantically enriched web services for the travel industry. *ACM Sigmod Record*, 33(3):21–27.
- [Dolin *et al.*, 2006] Dolin, R. H., Alschuler, L., Boyer, S., Beebe, C., Behlen, F. M., Biron, P. V., y Shabo, A. (2006). HL7 clinical document architecture, release 2. *Journal of the American Medical Informatics Association*, 13(1):30–39.
- [Dong y Chen, 2001] Dong, M. y Chen, F. F. (2001). Process modeling and analysis of manufacturing supply chain networks using object-oriented petri nets. *Robotics and Computer-Integrated Manufacturing*, 17(1):121–129.
- [Dubray *et al.*, 2006] Dubray, J.-J., Amand, S. S., y Martin, M. J. (2006). ebxml business process specification schema technical specification v2. 0.4. *Committee specification, UN/CEFACT*.
- [Eshlaghy *et al.*, 2009] Eshlaghy, A. T., Ghatari, A. R., Nikoomaram, H., y Zandhessami, H. (2009). Process based agile supply chain model according to bpr and idf 3.0 concepts. *Contemporary Engineering Sciences*, 2(3):117–138.
- [Esposito y Evangelista, 2014] Esposito, E. y Evangelista, P. (2014). Investigating virtual enterprise models: literature review and empirical findings. *International Journal of Production Economics*, 148:145–157.
- [Estefan *et al.*, 2009] Estefan, J. A., Laskey, K., McCabe, F. G., y Thornton, D. (2009). Reference architecture foundation for service oriented architecture. *Version*, 1:14.
- [Fabbe-Costes *et al.*, 2008] Fabbe-Costes, N., Jahre, M., y Roussat, C. (2008). Supply chain integration: the role of logistics service providers. *International Journal of Productivity and Performance Management*, 58(1):71–91.
- [Fawcett y Magnan, 2002] Fawcett, S. E. y Magnan, G. M. (2002). The rhetoric and reality of supply chain integration. *International Journal of Physical Distribution & Logistics Management*, 32(5):339–361.
- [Fliedner, 2003] Fliedner, G. (2003). Cpfr: an emerging supply chain tool. *Industrial Management & data systems*, 103(1):14–21.
- [Fontoura *et al.*, 2000] Fontoura, M., Pree, W., y Rumpe, B. (2000). *The UML profile for framework architectures*. Addison-Wesley Longman Publishing Co., Inc.
- [Framework, 2009] Framework, S. G. (2009). Technical standard. *The Open Group*.

- [Friedenthal *et al.*, 2014] Friedenthal, S., Moore, A., y Steiner, R. (2014). *A practical guide to SysML: the systems modeling language*. Morgan Kaufmann.
- [Frohlich y Westbrook, 2001] Frohlich, M. T. y Westbrook, R. (2001). Arcs of integration: an international study of supply chain strategies. *Journal of operations management*, 19(2):185–200.
- [García-Borgoñon, 2016] García-Borgoñon, L. (2016). *Un marco de referencia para facilitar la interoperabilidad y mantenibilidad de los modelos de procesos de software*. Tesis doctoral, Universidad de Sevilla.
- [García-Borgoñon *et al.*, 2014] García-Borgoñon, L., Barcelona, M., García-García, J., Alba, M., y Escalona, M. J. (2014). Software process modeling languages: A systematic literature review. *Information and Software Technology*, 56(2):103–116.
- [García García, 2015] García García, J. A. (2015). Una propuesta para el uso del paradigma guiado por modelos (mde) para la definición y ejecución de procesos de negocios.
- [Gardner *et al.*, 2003] Gardner, T., Griffin, C., Koehler, J., y Hauser, R. (2003). A review of OMG MOF 2.0 Query/Views/Transformations Submissions and Recommendations towards the final Standard. En *MetaModelling for MDA Workshop*, pp. 178–197. Citeseer.
- [Ghezzi, 2005] Ghezzi, C. (2005). Service-oriented computing: Where does it come from? a software engineering perspective. En *keynote address, 3rd International Conf. Service-Oriented Computing, Amsterdam*.
- [Gong *et al.*, 2006] Gong, R., Li, Q., Ning, K., Chen, Y., y O’ullivan, D. (2006). Business process collaboration using semantic interoperability: Review and framework. En *The Semantic Web—ASWC 2006*, pp. 191–204. Springer.
- [Grefen, 2009] Grefen, P. (2009). Systems for interorganizational business process management. *Handbook of research on business process modeling*, p. 403.
- [Grönroos, 1994] Grönroos, C. (1994). Quo vadis, marketing? toward a relationship marketing paradigm. *Journal of marketing management*, 10(5):347–360.
- [2007] OM Group and others (2007). Software process engineering metamodel (spem) 2.0 on omg.
- [Gruat La Forme *et al.*, 2007] Gruat La Forme, F.-A., Genoulaz, V. B., y Campagne, J.-P. (2007). A framework to analyse collaborative performance. *Computers in Industry*, 58(7):687–697.
- [Grubic y Fan, 2010] Grubic, T. y Fan, I.-S. (2010). Supply chain ontology: Review, analysis and synthesis. *Computers in Industry*, 61(8):776–786.
- [Hammer y Champy, 1993] Hammer, M. y Champy, J. (1993). Reengineering the corporation: A manifesto for business revolution. *Business horizons*, 36(5):90–91.
- [Hammer y Mangurian, 1990] Hammer, M. y Mangurian, G. E. (1990). The changing value of communications technology. En *Management information systems*, pp. 50–60. Scott, Foresman & Co.

- [Harding, 2008] Harding, C. (2008). Service-oriented architecture ontology. the open group draft 2.0.
- [Harmon, 2003] Harmon, P. (2003). Second generation business process methodologies. *Business Process Trends*, 1(5):1–12.
- [Hauder *et al.*, 2014] Hauder, M., Pigat, S., y Matthes, F. (2014). Research challenges in adaptive case management: A literature review. En *Enterprise Distributed Object Computing Conference Workshops and Demonstrations (EDOCW), 2014 IEEE 18th International*, pp. 98–107. IEEE.
- [Heidari *et al.*, 2013] Heidari, F., Loucopoulos, P., Brazier, F., y Barjis, J. (2013). A meta-meta-model for seven business process modeling languages. En *Business Informatics (CBI), 2013 IEEE 15th Conference on*, pp. 216–221. IEEE.
- [Herrmann y Kurz, 2011] Herrmann, C. y Kurz, M. (2011). Adaptive case management: supporting knowledge intensive processes with it systems. En *International Conference on Subject-Oriented Business Process Management*, pp. 80–97. Springer.
- [Herzberg *et al.*, 2014] Herzberg, N., Kirchner, K., y Weske, M. (2014). Modeling and monitoring variability in hospital treatments: A scenario using cmmn. En *International Conference on Business Process Management*, pp. 3–15. Springer.
- [Hofreiter *et al.*, 2006] Hofreiter, B., Huemer, C., Liegl, P., Schuster, R., y Zapletal, M. (2006). Un/cefact's modeling methodology (umm): a uml profile for b2b e-commerce. En *International Conference on Conceptual Modeling*, pp. 19–31. Springer.
- [Hofreiter, 2009] Hofreiter, B. b. (2009). Extending un/cefact's modeling methodology by a uml profile for local choreographies. *Information Systems and e-Business Management*, 7(2):251–271. cited By 2.
- [Holweg *et al.*, 2005] Holweg, M., Disney, S., Holmström, J., y Småros, J. (2005). Supply chain collaboration:: Making sense of the strategy continuum. *European management journal*, 23(2):170–181.
- [Hongxiu *et al.*, 2007] Hongxiu, W., Gang, W., Xiaoxian, W., y Guoan, G. (2007). Business process modeling for multi-enterprise cooperation.
- [Huan *et al.*, 2004] Huan, S. H., Sheoran, S. K., y Wang, G. (2004). A review and analysis of supply chain operations reference (scor) model. *Supply Chain Management: An International Journal*, 9(1):23–29.
- [Hübner, 2008] Hübner, U. (2008). Achieving interorganizational connectivity. En *eBusiness in Healthcare*, pp. 103–125. Springer.
- [Huhns *et al.*, 2002] Huhns, M. N., Stephens, L. M., e Ivezic, N. (2002). Automating supply-chain management. En *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 3*, pp. 1017–1024. ACM.
- [Hull *et al.*, 2010] Hull, Richard and Damaggio, Elio and Fournier, Fabiana and Gupta, Manmohan and Heath III, Fenno Terry and Hobson, Stacy and Linehan, Mark and Maradugu,

- Sridhar and Nigam, Anil and Sukaviriya, Piyawadee and others (2010). Introducing the guard-stage-milestone approach for specifying business entity lifecycles. En *Web services and formal methods*, pp. 1–24. Springer.
- [Hull *et al.*, 2013] Hull, R., Su, J., y Vaculin, R. (2013). Data management perspectives on business process management: tutorial overview. En *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*, pp. 943–948. ACM.
- [IKV++, 2011] IKV++ (2011). QVTMedini.
- [Institute for Supply Management, 2009] Institute for Supply Management (2009). ISM Glossary of Key Supply Management Terms, Fifth Edition.
- [Isik *et al.*, 2013] Isik, Ö., Mertens, W., y Van den Bergh, J. (2013). Practices of knowledge intensive process management: quantitative insights. *Business Process Management Journal*, 19(3):515–534.
- [Ivanov y Sokolov, 2010] Ivanov, D. y Sokolov, B. (2010). Challenges in research on modern and future supply chains. *Adaptive Supply Chain Management*, pp. 57–67.
- [IWT2 y Consejería de Cultura de la Junta de Andalucía, 2013] IWT2 y Consejería de Cultura de la Junta de Andalucía (2013). Comparativa de herramientas de modelado. Proyecto de Calidad.
- [Jakjoud *et al.*, 2014] Jakjoud, A. b., Zrikem, M., Baron, C. c. d., y Ayadi, A. (2014). Syspem: Toward a consistent and unified system process engineering metamodel. *Journal of Intelligent Manufacturing*. cited By 0; Article in Press.
- [2010] Ján, Závadský and others (2010). Simulation and its purpose in implementing of business process management. *Advances In Management*.
- [Janvier-James, 2012] Janvier-James, A. M. (2012). A new introduction to supply chains and supply chain management: Definitions and theories perspective. *International Business Research*, 5(1):p194.
- [Jardim-Goncalves *et al.*, 2013] Jardim-Goncalves, R., Agostinho, C., Sarraipa, J., Grilo, A., y Mendonça, J. P. (2013). Reference framework for enhanced interoperable collaborative networks in industrial organisations.
- [Jouault *et al.*, 2008] Jouault, F., Allilaire, F., Bézivin, J., y Kurtev, I. (2008). ATL: A model transformation tool. *Science of Computer Programming*, 72(1):31–39.
- [Jouault y Kurtev, 2006] Jouault, F. y Kurtev, I. (2006). Transforming models with ATL. En *Satellite Events at the MoDELS 2005 Conference*, pp. 128–138. Springer.
- [Kaabi *et al.*, 2004] Kaabi, R. S., Souveyet, C., y Rolland, C. (2004). Eliciting service composition in a goal driven manner. En *Proceedings of the 2nd international conference on Service oriented computing*, pp. 308–315. ACM.
- [Kamoun, 2007] Kamoun, F. (2007). A roadmap towards the convergence of business process management and service oriented architecture. *Ubiquity*, 2007(April):2.

- [Kanda *et al.*, 2008] Kanda, Arun and Deshmukh, SG and others (2008). Supply chain coordination: perspectives, empirical studies and research directions. *International journal of production Economics*, 115(2):316–335.
- [Kemppainen y Vepsäläinen, 2007] Kemppainen, K. y Vepsäläinen, A. P. (2007). Logistical and technological differentiation as a precondition of supply networking. *International Journal of Logistics Management, The*, 18(1):81–101.
- [Kim y Rogers, 2005] Kim, J. y Rogers, K. (2005). An object-oriented approach for building a flexible supply chain model. *International Journal of Physical Distribution & Logistics Management*, 35(7):481–502.
- [Kim *et al.*, 2013] Kim, Y., Lee, Y., Chung, K.-Y., y Lee, K.-D. (2013). An investigation on the information systems research in supply chain management: an analysis of research topic and methodology. *Multimedia Tools and Applications*, pp. 1–12.
- [Kitchenham y Brereton, 2013] Kitchenham, B. y Brereton, P. (2013). A systematic review of systematic review process research in software engineering. *Information and Software Technology*, 55(12):2049 – 2075.
- [Kitchenham y Charters, 2007] Kitchenham, B. y Charters, S. (2007). Guidelines for performing Systematic Literature Reviews in Software Engineering. Technical Report EBSE 2007-001, Keele University and Durham University Joint Report.
- [Kitchenham *et al.*, 2010] Kitchenham, B., Pretorius, R., Budgen, D., Pearl Brereton, O., Turner, M., Niazi, M., y Linkman, S. (2010). Systematic literature reviews in software engineering—a tertiary study. *Information and Software Technology*, 52(8):792–805.
- [Kleppe *et al.*, 2003] Kleppe, A. G., Warmer, J. B., y Bast, W. (2003). *MDA explained: the model driven architecture: practice and promise*. Addison-Wesley Professional.
- [Kolovos *et al.*, 2008] Kolovos, D. S., Paige, R. F., y Polack, F. A. (2008). The epsilon transformation language. En *Theory and practice of model transformations*, pp. 46–60. Springer.
- [Korherr y List, 2006] Korherr, B. y List, B. (2006). A uml 2 profile for event driven process chains. En *Research and Practical Issues of Enterprise Information Systems*, pp. 161–172. Springer.
- [Kosanke, 2009] Kosanke, K. (2009). Business process modelling and standardisation.
- [Kotler, 2011] Kotler, P. (2011). Reinventing marketing to manage the environmental imperative. *Journal of Marketing*, 75(4):132–135.
- [Kreger y Estefan, 2009] Kreger, H. y Estefan, J. (2009). Navigating the soa open standards landscape around architecture. *Joint Paper, The Open Group, OASIS, and OMG*.
- [Künzle y Reichert, 2011] Künzle, V. y Reichert, M. (2011). Philharmonicflows: towards a framework for object-aware process management. *Journal of Software Maintenance and Evolution: Research and Practice*, 23(4):205–244.

- [Kurz *et al.*, 2015] Kurz, M., Schmidt, W., Fleischmann, A., y Lederer, M. (2015). Leveraging cmmn for acm: examining the applicability of a new omg standard for adaptive case management. En *Proceedings of the 7th International Conference on Subject-Oriented Business Process Management*, p. 4. ACM.
- [La Rosa *et al.*, 2013] La Rosa, M., van der Aalst, W. M., Dumas, M., y Milani, F. P. (2013). Business process variability modeling: A survey.
- [Lambert, 2008] Lambert, D. M. (2008). An executive summary of supply chain management: Processes. *Partnerships, Performance. Supply Chain Management Institute. Sarasota, FL.*
- [Lambert *et al.*, 1998] Lambert, D. M., Cooper, M. C., y Pagh, J. D. (1998). Supply chain management: implementation issues and research opportunities. *International Journal of Logistics Management, The*, 9(2):1–20.
- [Lambert *et al.*, 2005] Lambert, D. M., García-Dastugue, S. J., y Croxton, K. L. (2005). An evaluation of process-oriented supply chain management frameworks. *Journal of business Logistics*, 26(1):25–51.
- [Lapide, 2006] Lapide, L. (2006). Top-down & bottom-up forecasting in s&op. *The Journal of Business Forecasting*, 25(2):14.
- [Larson y Halldorsson, 2004] Larson, P. D. y Halldorsson, A. (2004). Logistics versus supply chain management: an international survey. *International Journal of Logistics: Research and Applications*, 7(1):17–31.
- [Lave y Wenger, 1991] Lave, J. y Wenger, E. (1991). *Situated learning: Legitimate peripheral participation*. Cambridge university press.
- [Legner y Wende, 2007] Legner, C. y Wende, K. (2007). The challenges of inter-organizational business process design—a research agenda.
- [Leymann, 2003] Leymann, F. (2003). Web services: Distributed applications without limits. En *BTW*, volumen 26, pp. 2–23. Citeseer.
- [Leymann, 2005] Leymann, F. (2005). Combining web services and the grid: Towards adaptive enterprise applications. En *CAiSE Workshops (2)*, pp. 9–21.
- [Leymann y Roller, 2000] Leymann, F. y Roller, D. (2000). Production workflow: concepts and techniques.
- [Li *et al.*, 2008] Li, L., Hosking, J., y Grundy, J. (2008). Maramaeml: An integrated multi-view business process modelling environment with tree-overlays, zoomable interfaces and code generation. En *Proceedings of the 2008 23rd IEEE/ACM International Conference on Automated Software Engineering*, pp. 477–478. IEEE Computer Society.
- [Li *et al.*, 2009] Li, L., Zhu, J.-j., y Li, H.-m. (2009). Idef-based construction supply chain management. En *2009 First International Conference on Information Science and Engineering*, pp. 4899–4902. IEEE.
- [Li *et al.*, 2007] Li, R. L., Hosking, J. G., y Grundy, J. C. (2007). Eml: A tree overlay-based visual language for business process modelling. En *ICEIS (3)*, pp. 131–137. Citeseer.

- [Lipnack, 1997] Lipnack, J. (1997). *Virtual teams: Reaching across space, time, and organizations with technology*. Jeffrey Stamps.
- [Lockamy III y McCormack, 2004] Lockamy III, A. y McCormack, K. (2004). Linking score planning practices to supply chain performance: An exploratory study. *International Journal of Operations & Production Management*, 24(12):1192–1218.
- [Lucas *et al.*, 2005] Lucas, P., Hommerson, A., Galan, J., Marcos, M., Coltell, O., Mouzon, O., Polo, C., Rosenbrand, K., Wittenberg, J., y van Croonenborg, J. (2005). New model of guideline process. specific targeted research project information society technology, universitat jaume i. Technical report, IST-FP6-508794.
- [Malekan y Afsarmanesh, 2014] Malekan, H. S. y Afsarmanesh, H. (2014). Overview of business process modeling languages supporting enterprise collaboration. En *Business Modeling and Software Design*, pp. 24–45. Springer.
- [Marin *et al.*, 2013] Marin, M., Hull, R., y Vaculín, R. (2013). Data centric bpm and the emerging case management standard: A short survey. En *Business Process Management Workshops*, pp. 24–30. Springer.
- [Marin y Brown, 2015] Marin, M. A. y Brown, J. A. (2015). Implementing a case management modeling and notation (cmmn) system using a content management interoperability services (cmis) compliant repository. *arXiv preprint arXiv:1504.06778*.
- [Marin *et al.*, 2015] Marin, M. A., Lotriet, H., y Van Der Poll, J. A. (2015). Metrics for the case management modeling and notation (cmmn) specification. En *Proceedings of the 2015 Annual Research Conference on South African Institute of Computer Scientists and Information Technologists*, p. 28. ACM.
- [Mayer *et al.*, 1995] Mayer, R. J., Menzel, C. P., Painter, M. K., Dewitte, P. S., Blinn, T., y Perakath, B. (1995). Information integration for concurrent engineering (iice) idef3 process description capture method report. Technical report, DTIC Document.
- [Mellor, 2004] Mellor, S. J. (2004). *MDA distilled: principles of model-driven architecture*. Addison-Wesley Professional.
- [Mens y Van Gorp, 2006] Mens, T. y Van Gorp, P. (2006). A taxonomy of model transformation. *Electronic Notes in Theoretical Computer Science*, 152:125–142.
- [Mentzer *et al.*, 2001] Mentzer, J. T., DeWitt, W., Keebler, J. S., Min, S., Nix, N. W., Smith, C. D., y Zacharia, Z. G. (2001). Defining supply chain management. *Journal of Business logistics*, 22(2):1–25.
- [Mili *et al.*, 2010] Mili, H., Tremblay, G., Jaoude, G. B., Lefebvre, É., Elabed, L., y Boussaidi, G. E. (2010). Business process modeling languages: Sorting through the alphabet soup. *ACM Computing Surveys (CSUR)*, 43(1):4.
- [Min y Zhou, 2002] Min, H. y Zhou, G. (2002). Supply chain modeling: past, present and future.
- [Miniet y Navarro, 2014] Miniet, Y. V. y Navarro, M. R. (2014). Modelado del proceso de desarrollo de software con el modelo y notación de gestión de casos (cmmn).

- [Mohagheghi *et al.*, 2013] Mohagheghi, P., Gilani, W., Stefanescu, A., y Fernandez, M. A. (2013). An empirical study of the state of the practice and acceptance of model-driven engineering in four industrial cases.
- [Morash y Clinton, 1998] Morash, E. A. y Clinton, S. R. (1998). Supply chain integration: customer value through collaborative closeness versus operational excellence. *Journal of Marketing Theory and Practice*, pp. 104–120.
- [Motahari-Nezhad y Swenson, 2013] Motahari-Nezhad, H. R. y Swenson, K. D. (2013). Adaptive case management: Overview and research challenges. En *Business Informatics (CBI), 2013 IEEE 15th Conference on*, pp. 264–269. IEEE.
- [Muller *et al.*, 2005] Muller, P.-A., Fleurey, F., y Jézéquel, J.-M. (2005). Weaving executability into object-oriented meta-languages. En *Model Driven Engineering Languages and Systems*, pp. 264–278. Springer.
- [Naslund y Williamson, 2010] Naslund, D. y Williamson, S. (2010). What is management in supply chain management?-a critical review of definitions, frameworks and terminology. *Journal of Management Policy and Practice*, 11(4):11–28.
- [Newman *et al.*, 2009] Newman, W. R., Hanna, M. D., Gattiker, T., y Huang, X. (2009). Charting supply chain management integration and initiatives: A framework to guide implementation. *American Journal of Business*, 24(1):19–32.
- [Nigam y Caswell, 2003] Nigam, A. y Caswell, N. S. (2003). Business artifacts: An approach to operational specification. *IBM Systems Journal*, 42(3):428–445.
- [Noordin *et al.*, 2012] Noordin, N. A., Bititci, U. S., y Van Der Meer, R. (2012). Review on collaborative decision making in supply chain: The relationship between e-collaboration technology and development of inter-organizational trust.
- [OASIS, 2015] OASIS (2015). Oasis website.
- [Object Management Group, 2009] Object Management Group (2009). Case Management Process Modeling Request For Proposal, OMG Document: Bmi 2009-09-23. [Online; accessed 22-Dec-2015].
- [Oliva *et al.*, 2011] Oliva, G., Hatori, F., Leite, L., y Gerosa, M. (2011). Web services choreographies adaptation: A systematic review.
- [Oliver y Webber, 1982] Oliver, R. K. y Webber, M. D. (1982). Supply-chain management: logistics catches up with strategy.
- [OMG, 2004] OMG (2004). UML Profile for Enterprise Distributed Object Computing. [Online; accessed 8-June-2016].
- [OMG, 2010] OMG (2010). Business Process Model and Notation (BPMN) Version 2.0. Object Management Group. [Online; accessed 3-July-2014].
- [OMG, 2011a] OMG (2011a). Meta Object Facility (MOF) 2.0 Query/View/Transformation Specification (QVT) Version 1.1. Object Management Group. [Online; accessed 18-July-2014].

- [OMG, 2011b] OMG (2011b). Unified modeling language 2.4.1 specification, <http://www.omg.org/spec/uml/2.4.1/>, accessed 2013-04-05.
- [OMG, 2012] OMG (2012). Object Constraint Language (OCL) Core Version 2.3.1. Object Management Group. [Online; accessed 8-June-2014].
- [OMG, 2013] OMG (2013). Omg's metaobject facility.
- [OMG, 2014] OMG (2014). Case Management Model and Notation (CMMN), Version 1.0. [Online; accessed 12-December-2015].
- [OMG, 2015a] OMG (2015a). Object management group website.
- [OMG, 2015b] OMG (2015b). Unified Modeling Language (UML) - Superstructure. Version 2.5. Object Management Group. [Online; accessed 8-June-2016].
- [OMG, 2016] OMG (2016). Case Management Model and Notation (CMMN), Version Beta 1.1. [Online; accessed 4-August-2016].
- [OpenGroup, 2015] OpenGroup (2015). The open group website.
- [Panayides, 2006] Panayides, P. M. (2006). Maritime logistics and global supply chains: towards a research agenda. *Maritime Economics & Logistics*, 8(1):3–18.
- [Papazoglou, 2003] Papazoglou, M. P. (2003). Service-oriented computing: Concepts, characteristics and directions. En *Web Information Systems Engineering, 2003. WISE 2003. Proceedings of the Fourth International Conference on*, pp. 3–12. IEEE.
- [Papazoglou et al., 2008] Papazoglou, M. P., Traverso, P., Dustdar, S., y Leymann, F. (2008). Service-oriented computing: a research roadmap. *International Journal of Cooperative Information Systems*, 17(02):223–255.
- [Papazoglou y Van Den Heuvel, 2007a] Papazoglou, M. P. y Van Den Heuvel, W.-J. (2007a). Business process development life cycle methodology. *Communications of the ACM*, 50(10):79–85.
- [Papazoglou y Van Den Heuvel, 2007b] Papazoglou, M. P. y Van Den Heuvel, W.-J. (2007b). Service oriented architectures: approaches, technologies and research issues. *The VLDB journal*, 16(3):389–415.
- [Parazoglou, 2006] Parazoglou, M. P. (2006). *E-Business Organisational & Technical Foundations*. John Wiley & Sons.
- [Paszkievicz y Cellary, 2012] Paszkievicz, Z. y Cellary, W. (2012). Computer supported collaborative processes in virtual organizations. *arXiv preprint arXiv:1205.4653*.
- [Pawlewski, 2014] Pawlewski, P. (2014). Multimodal approach to modeling of manufacturing processes. *Procedia {CIRP}*, 17(0):716 – 720. Variety Management in Manufacturing Proceedings of the 47th {CIRP} Conference on Manufacturing Systems.
- [Penserini et al., 2006] Penserini, L., Perini, A., Susi, A., y Mylopoulos, J. (2006). From stakeholder needs to service requirements. En *Service-Oriented Computing: Consequences for Engineering Requirements, 2006. SOCCER'06*, pp. 8–8. IEEE.

- [Perego *et al.*, 2011] Perego, A., Perotti, S., y Mangiaracina, R. (2011). Ict for logistics and freight transportation: a literature review and research agenda.
- [Petri, 1962] Petri, C. A. (1962). Kommunikation mit automaten.
- [Picard, 2013] Picard, W. (2013). *Adaptation of service protocols*. Poznań University of Economics Press.
- [Picard *et al.*, 2014] Picard, W., Paszkiewicz, Z., Strykowski, S., Wojciechowski, R., y Cellary, W. (2014). Application of the service-oriented architecture at the inter-organizational level. En *Advanced SOA Tools and Applications*, pp. 125–201. Springer.
- [Pires y de Camargo Junior, 2010] Pires, S. R. y de Camargo Junior, J. B. (2010). Using cloud computing to integrate processes in the supply chain.
- [Rabe *et al.*, 2006] Rabe, M., Jaekel, F.-W., y Weinaug, H. (2006). Reference models for supply chain design and configuration. pp. 1143–1150. cited By 6.
- [Ragatz *et al.*, 2002] Ragatz, G. L., Handfield, R. B., y Petersen, K. J. (2002). Benefits associated with supplier integration into new product development under conditions of technology uncertainty. *Journal of Business Research*, 55(5):389–400.
- [Redding *et al.*, 2009] Redding, G., Dumas, M., Ter Hofstede, A. H., e Iordachescu, A. (2009). Modelling flexible processes with business objects. En *Commerce and Enterprise Computing, 2009. CEC'09. IEEE Conference on*, pp. 41–48. IEEE.
- [Reichert, 2012] Reichert, Manfred, W. B. (2012). *Enabling Flexibility in Process-Aware Information Systems*. Springer.
- [Riemer *et al.*, 2011] Riemer, K., Holler, J., e Indulska, M. (2011). Collaborative process modelling-tool analysis and design implications. En *ECIS*.
- [Romeikat *et al.*, 2008] Romeikat, R., Roser, S., Müllender, P., y Bauer, B. (2008). Translation of QVT relations into QVT operational mappings. En *Theory and Practice of Model Transformations*, pp. 137–151. Springer.
- [Roser *et al.*, 2006] Roser, S., Bauer, B., y Muller, J. P. (2006). Model-and architecture-driven development in the context of cross-enterprise business process engineering.
- [Sadiq *et al.*, 2005] Sadiq, S. W., Orlowska, M. E., y Sadiq, W. (2005). Specification and validation of process constraints for flexible workflows. *Information Systems*, 30(5):349–378.
- [SCC, 2014] SCC, A. (2014). *Supply Chain Operations Reference Model v11*.
- [Scheer *et al.*, 2005] Scheer, A.-W., Thomas, O., y Adam, O. (2005). Process modeling using event-driven process chains. *Process-Aware Information Systems*, pp. 119–146.
- [Schmidt, 2006] Schmidt, D. C. (2006). Model-driven engineering.
- [Schürr, 1995] Schürr, A. (1995). Specification of graph translators with triple graph grammars. pp. 151–163.
- [Sendall y Kozaczynski, 2003] Sendall, S. y Kozaczynski, W. (2003). Model transformation: The heart and soul of model-driven software development. *IEEE Software*, (20(5)):42–45.

- [services GmbH, 2016] services GmbH, C. (2016). Camunda tool.
- [Shapiro, 1999] Shapiro, J. F. (1999). *Bottom-Up Vs. Top-Down Approaches to Supply Chain Modeling*, pp. 737–759. Springer US, Boston, MA.
- [2000] Soley, Richard and others (2000). Model driven architecture.
- [SparxSystems, 2014] SparxSystems (2014). Enterprise Architect.
- [Stadtler, 2005] Stadtler, H. (2005). Supply chain management and advanced planning—basics, overview and challenges.
- [Standard, 2006] Standard, O. (2006). Reference model for service oriented architecture v1.0. *OASIS, October*.
- [Standard, 2008] Standard, O. (2008). Reference architecture for soa foundation v1.0. *OASIS, April*.
- [Stevens, 1990] Stevens, G. C. (1990). Successful supply-chain management. *Management Decision*, 28(8).
- [Stock, 2009] Stock, J. R. (2009). A research view of supply chain management: Developments and topics for exploration.
- [Stock y Boyer, 2009] Stock, J. R. y Boyer, S. L. (2009). Developing a consensus definition of supply chain management: a qualitative study. *International Journal of Physical Distribution & Logistics Management*, 39(8):690–711.
- [Supply Chain Management Institute, 2013] Supply Chain Management Institute (2013). SCMi Business Model. [Online; accessed 29-Oct-2014].
- [Supply Chain Management Institute (Douglas M. Lambert), 2008] Supply Chain Management Institute (Douglas M. Lambert) (2008). *Supply Chain Management: Processes, Partnerships, Performance, Third Edition*.
- [Sweeney, 2005] Sweeney, E. (2005). Perspectives on supply chain management and logistics definitions.
- [Swenson, 2011] Swenson, K. (2011). *Taming the Unpredictable: Real World Adaptive Case Management: Case Studies and Practical Guidance*. Future Strategies Inc.
- [Swenson et al., 2010] Swenson, Keith D and Palmer, Nathaniel and others (2010). *Mastering the unpredictable: How adaptive case management will revolutionize the way that knowledge workers get things done*, volumen 1. Meghan-Kiffer Press Tampa.
- [Taentzer, 2004] Taentzer, G. (2004). AGG: A graph transformation environment for modeling and validation of software. En *Applications of Graph Transformations with Industrial Relevance*, pp. 446–453. Springer.
- [Taleb, 2012] Taleb, N. N. (2012). *Anti-fragile: How to Live in a World We Don't Understand*, volumen 3. Allen Lane London, UK.
- [Tarhan et al., 2016] Tarhan, A., Turetken, O., y Reijers, H. A. (2016). Business process maturity models: a systematic literature review. *Information and Software Technology*, 75:122–134.

- [Taylor, 1914] Taylor, F. W. (1914). *The principles of scientific management*. Harper.
- [The Eclipse Foundation, 2014a] The Eclipse Foundation (2014a). BPMN2 Modeler.
- [The Eclipse Foundation, 2014b] The Eclipse Foundation (2014b). Eclipse Luna.
- [The Eclipse Foundation, 2014c] The Eclipse Foundation (2014c). EuGENia.
- [The Eclipse Foundation, 2015] The Eclipse Foundation (2015). Papyrus.
- [Thiry y Thirion, 2009] Thiry, L. y Thirion, B. (2009). Functional metamodels for systems and software. *Journal of Systems and Software*, 82(7):1125–1136.
- [Tounsi *et al.*, 2012] Tounsi, J., Habchi, G., Boissière, J., y Azaiez, S. (2012). A multi-agent knowledge model for smes mechatronic supply chains.
- [Vaculin *et al.*, 2011] Vaculin, R., Hull, R., Heath, T., Cochran, C., Nigam, A., y Sukaviriya, P. (2011). Declarative business artifact centric modeling of decision and knowledge intensive business processes. En *Enterprise Distributed Object Computing Conference (EDOC), 2011 15th IEEE International*, pp. 151–160. IEEE.
- [Van der Aalst *et al.*, 2003] Van der Aalst, W., Stoffele, M., y Wamelink, J. (2003). Case handling in construction. *Automation in Construction*, 12(3):303–320.
- [Van der Aalst *et al.*, 2010] Van der Aalst, W. M., Nakatumba, J., Rozinat, A., y Russell, N. (2010). Business process simulation. En *Handbook on Business Process Management 1*, pp. 313–338. Springer.
- [Van der Aalst *et al.*, 2005] Van der Aalst, W. M., Weske, M., y Grünbauer, D. (2005). Case handling: a new paradigm for business process support. *Data & Knowledge Engineering*, 53(2):129–162.
- [Van Lamsweerde, 2000] Van Lamsweerde, A. (2000). Requirements engineering in the year 00: A research perspective. En *Proceedings of the 22nd international conference on Software engineering*, pp. 5–19. ACM.
- [Varró *et al.*, 2002] Varró, D., Varró, G., y Pataricza, A. (2002). Designing the automatic transformation of visual languages. *Science of Computer Programming*, 44(2):205–227.
- [Vernadat, 2002] Vernadat, F. (2002). Ueml: towards a unified enterprise modelling language. *International Journal of Production Research*, 40(17):4309–4321.
- [Vo *et al.*, 2015] Vo, T. T., Coulette, B., Tran, H. N., y Lbath, R. (2015). Defining and using collaboration patterns for software process development. En *Model-Driven Engineering and Software Development (MODELSWARD), 2015 3rd International Conference on*, pp. 557–564. IEEE.
- [Wand y Weber, 2002] Wand, Y. y Weber, R. (2002). Research commentary: information systems and conceptual modeling? research agenda. *Information Systems Research*, 13(4):363–376.

- [Wang *et al.*, 2011] Wang, W., Chai, Y., Dong, J., Ding, H., Ren, C., y Qiu, M. (2011). Evaluating the value of collaboration in supply chain through business process simulation. En *Service Operations, Logistics, and Informatics (SOLI), 2011 IEEE International Conference on*, pp. 307–312. IEEE.
- [Webster Jr, 1992] Webster Jr, F. E. (1992). The changing role of marketing in the corporation. *The Journal of Marketing*, pp. 1–17.
- [Wen *et al.*, 2007] Wen, C., Li, X., y Bai, Y. (2007). Research on dynamic supply chain integration network model based on collaboration theory and non-linear polya processes. En *Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on*, pp. 6091–6094. IEEE.
- [Weske, 2010] Weske, M. (2010). *Business process management: concepts, languages, architectures*. Springer Publishing Company, Incorporated.
- [Wirth, 1971] Wirth, N. (1971). Program development by stepwise refinement. *Communications of the ACM*, 14(4):221–227.
- [Wohlin y Prikladniki, 2013] Wohlin, C. y Prikladniki, R. (2013). Systematic Literature Reviews in Software Engineering. *Information and Software Technology*, (0).
- [Xia *et al.*, 2006] Xia, R., Jiang, W., y Jian-ping, L. (2006). Construction of the strategic alliance of supply chain based on idf models [j]. *Journal of Kunming University of Science and Technology (Science and Technology)*, 4:025.
- [Xiaodong *et al.*, 2009] Xiaodong, W., Hongming, C., y Boyi, X. (2009). An extended petri-net based approach for supply chain process modeling and web service transformation. cited By 0.
- [Xu *et al.*, 2011] Xu, W., Su, J., Yan, Z., Yang, J., y Zhang, L. (2011). An artifact-centric approach to dynamic modification of workflow execution. En *On the Move to Meaningful Internet Systems: OTM 2011*, pp. 256–273. Springer.
- [Zailani y Rajagopal, 2005] Zailani, S. y Rajagopal, P. (2005). Supply chain integration and performance: Us versus east asian companies. *Supply Chain Management: An International Journal*, 10(5):379–393.
- [Zhang, 2010] Zhang, F. (2010). *Support for Dynamic Supply Chain Modelling*. Tesis doctoral, Computer Science)–University of Auckland.
- [Zhang y Ali Babar, 2013] Zhang, H. y Ali Babar, M. (2013). Systematic reviews in software engineering: An empirical investigation. *Information and Software Technology*, 55(7):1341–1354.
- [Zhang *et al.*, 2011] Zhang, H., Babar, M. A., y Tell, P. (2011). Identifying relevant studies in software engineering. *Information and Software Technology*, 53(6):625–637.
- [Zhang *et al.*, 2009] Zhang, L., You, X., Jiao, J., y Helo, P. (2009). Supply chain configuration with co-ordinated product, process and logistics decisions: an approach based on petri nets. *International journal of production Research*, 47(23):6681–6706.

- [Zhang *et al.*, 2010] Zhang, R., Hosking, J., Grundy, J., Mehandjiev, N., y Carpenter, M. (2010). Design of a suite of visual languages for supply chain specification. pp. 240–243. cited By 0.
- [Zhou y Rong, 2010] Zhou, K. y Rong, G. (2010). Study of supply chain monitoring system based on idf method. En *Logistics Systems and Intelligent Management, 2010 International Conference on*, volumen 1, pp. 278–281. IEEE.
- [Ziemann, 2010] Ziemann, J. (2010). *Architecture of Interoperable Information Systems: An Enterprise Model-Based Approach for Describing and Enacting Collaborative Business Processes*, volumen 23. Logos Verlag Berlin GmbH.
- [Ziemann *et al.*, 2007] Ziemann, J., Kahl, T., y Werth, D. (2007). Using view process models in collaborative business processes. *Encyclopedia of Networked and Virtual Organizations*, Idea Group Inc., Hershey *et al.*

Anexo A

Transformaciones M2M entre INROMA y CBGProcess

En este anexo se presentan las relaciones para acometer la transformación bidireccional y horizontal entre INROMA y CBGProcess. Para ello, haremos uso de la sintaxis concreta gráfica de QVT Relations, tal y como fueron descritas en el capítulo 6.

La tabla A.1 expone las correspondencias entre las relaciones y los diagramas de transformación QVT que se muestran a continuación.

Relación	Sintaxis Gráfica QVT Relacional
Process2Process	Figura A.1
MapActivities	Figura A.2
MapIndicators	Figura A.3
MapInitial	Figura A.4
MapFinal	Figura A.5
MapConditional	Figura A.6
MapInputProductsOfActivity	Figura A.7
MapOutputProductsOfActivity	Figura A.8
MapStakeholdersPartOfActivity	Figura A.9
MapStakeholdersRespOfActivity	Figura A.10
MapIncomingSequence	Figura A.11
MapOutgoingSequence	Figura A.12
MapMetrics	Figura A.13

Tabla A.1: Relaciones para acometer la transformación INROMA2CBGProcess

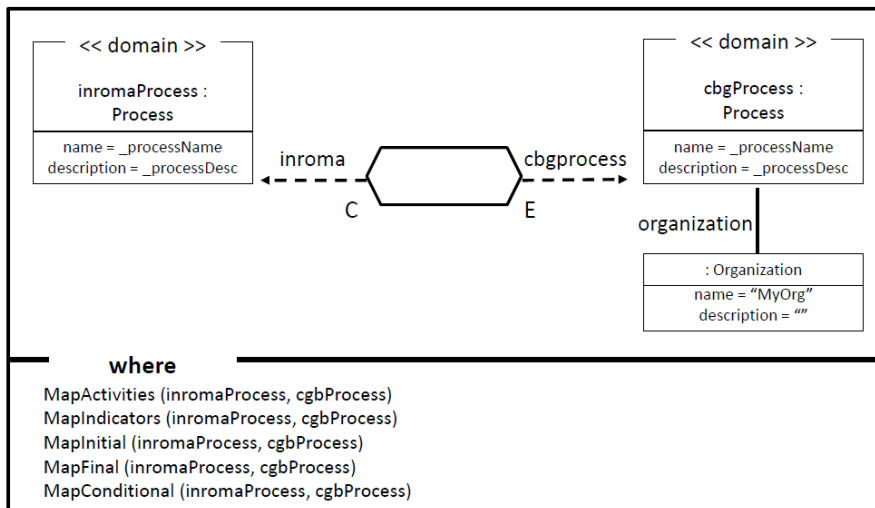
Process2Process

Figura A.1: Vista QVT Relations de la relación Process2Process en la transformación M2M inroma2CBGProcess

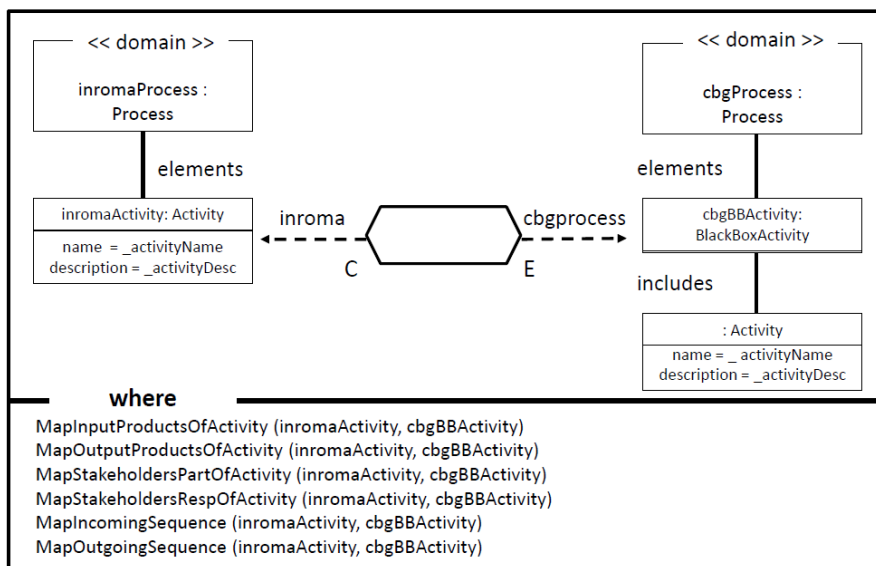
MapActivities

Figura A.2: Vista QVT Relations de la relación MapActivities en la transformación M2M inroma2CBGProcess

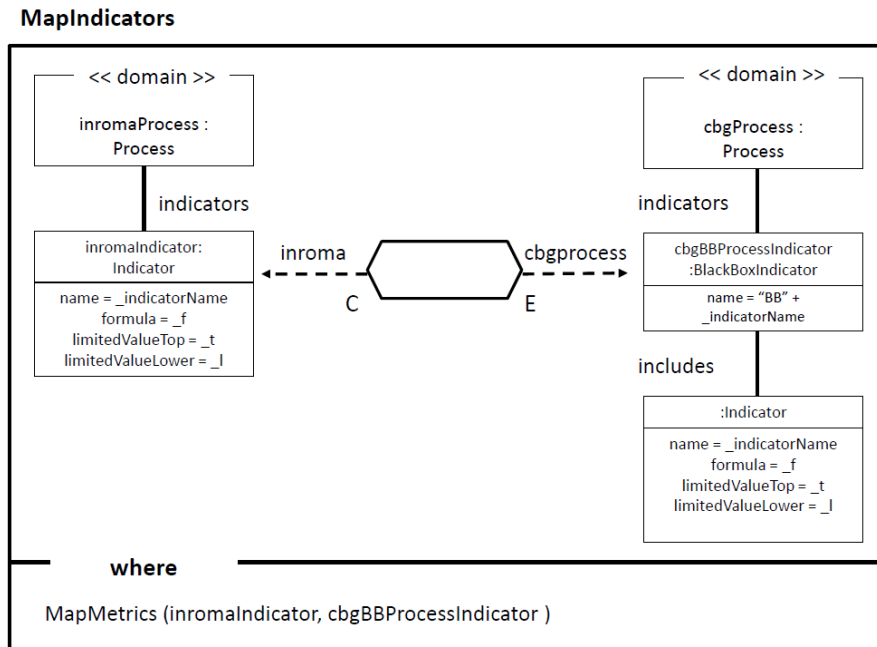


Figura A.3: Vista QVT Relations de la relación MapIndicators en la transformación M2M inroma2CBGProcess

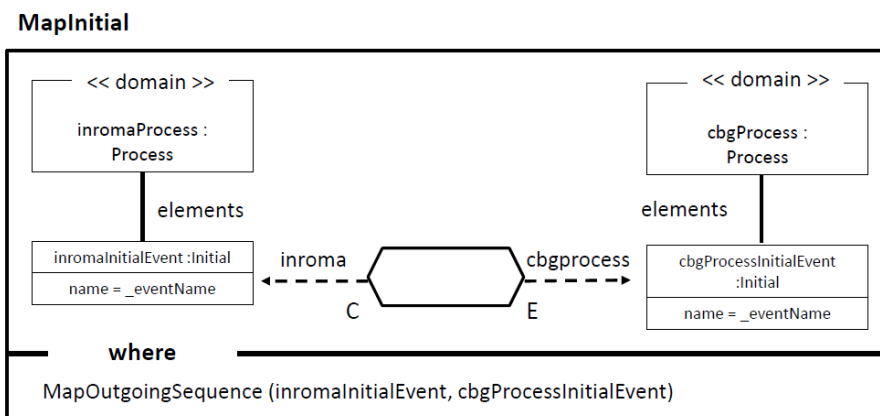


Figura A.4: Vista QVT Relations de la relación MapInitial en la transformación M2M inroma2CBGProcess

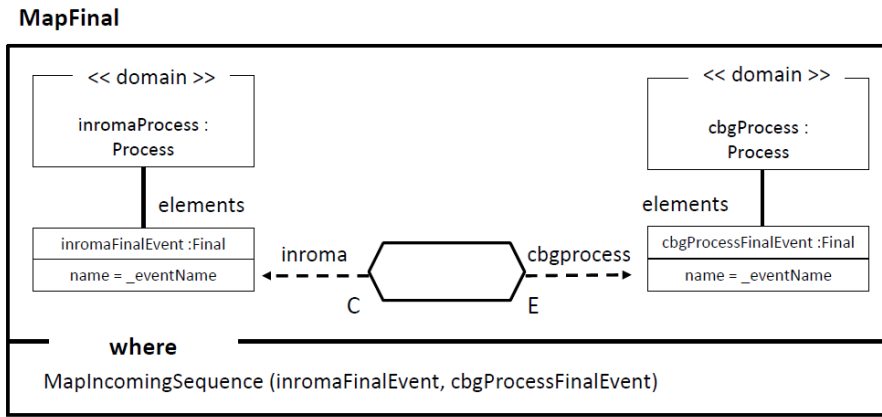


Figura A.5: Vista QVT Relations de la relación MapFinal en la transformación M2M inroma2CBGProcess

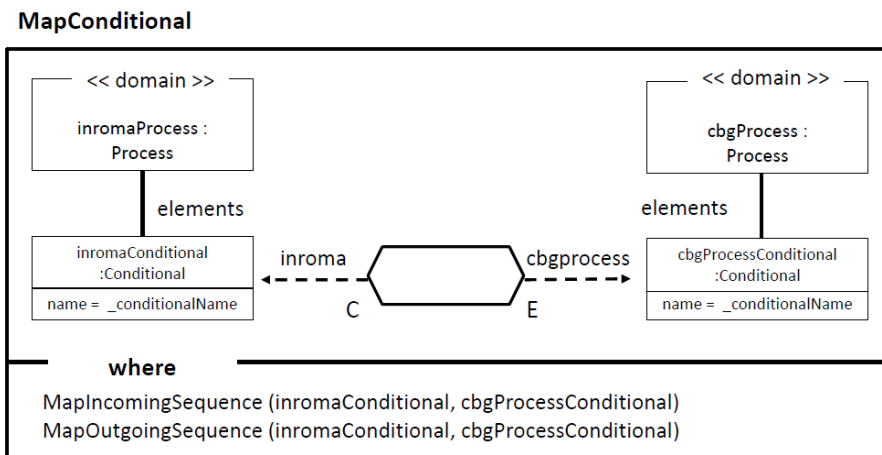


Figura A.6: Vista QVT Relations de la relación MapConditional en la transformación M2M inroma2CBGProcess

MapInputProductsOfActivity

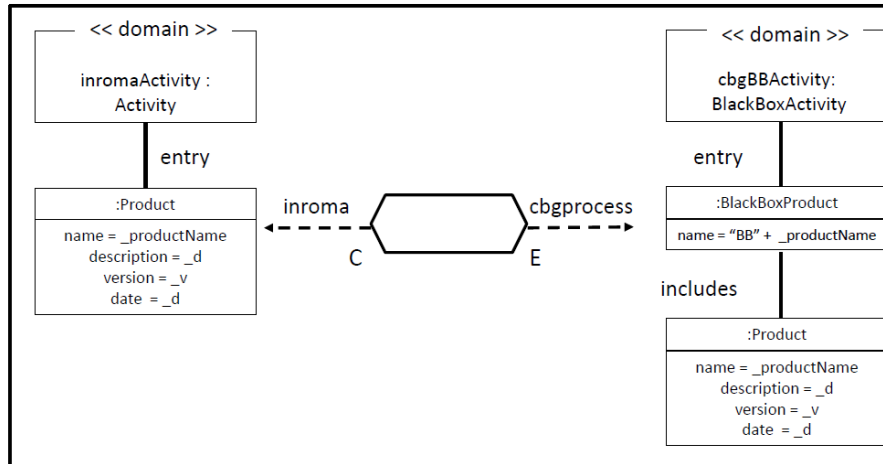


Figura A.7: Vista QVT Relations de la relación MapInputProductsOfActivity en la transformación M2M inroma2CBGProcess

MapOutputProductsOfActivity

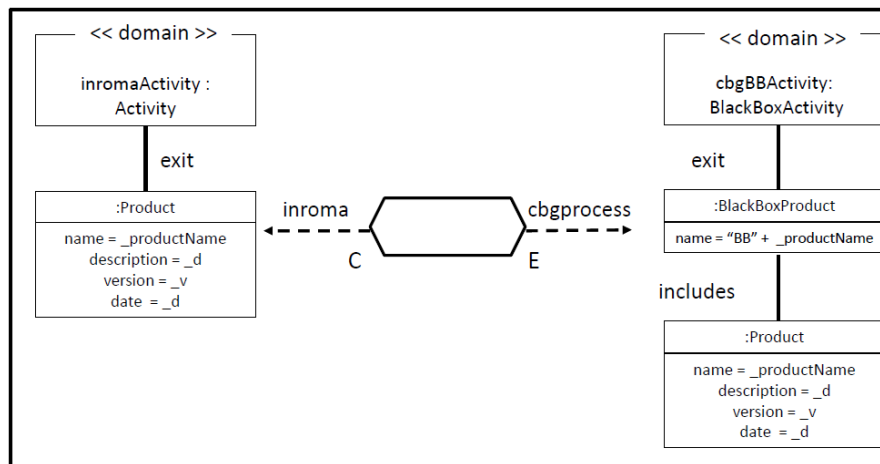


Figura A.8: Vista QVT Relations de la relación MapOutputProductsOfActivity en la transformación M2M inroma2CBGProcess

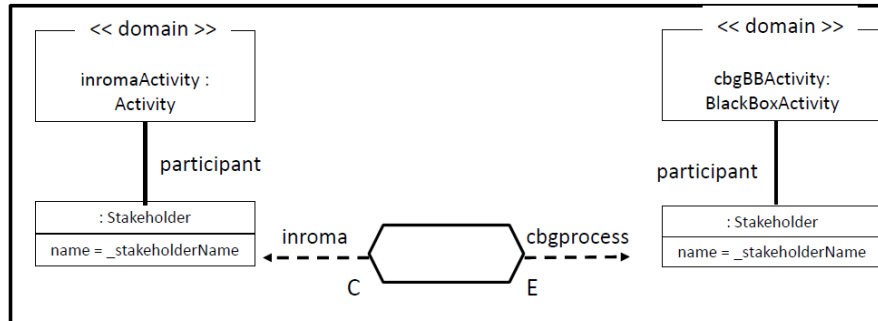
MapStakeholdersPartOfActivity

Figura A.9: Vista QVT Relations de la relación **MapStakeholdersPartOfActivity** en la transformación M2M **inroma2CBGProcess**

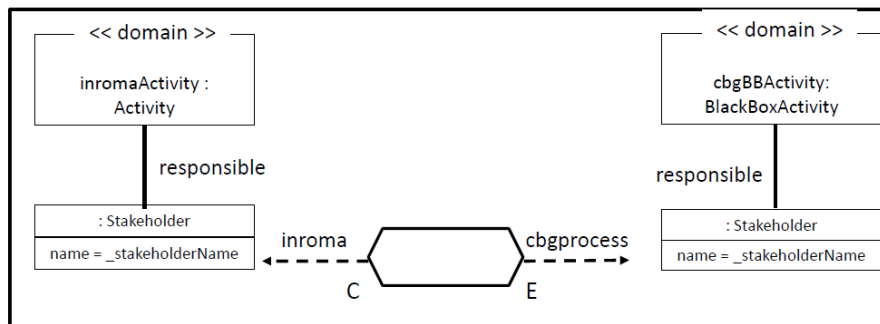
MapStakeholdersRespOfActivity

Figura A.10: Vista QVT Relations de la relación **MapStakeholdersRespOfActivity** en la transformación M2M **inroma2CBGProcess**

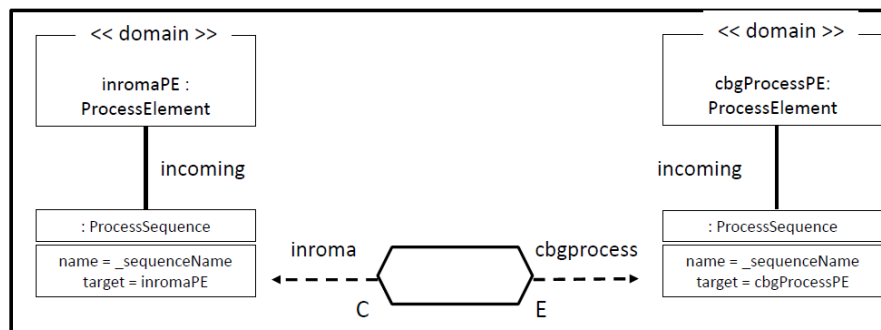
MapIncomingSequence

Figura A.11: Vista QVT Relations de la relación **MapIncomingSequence** en la transformación M2M **inroma2CBGProcess**

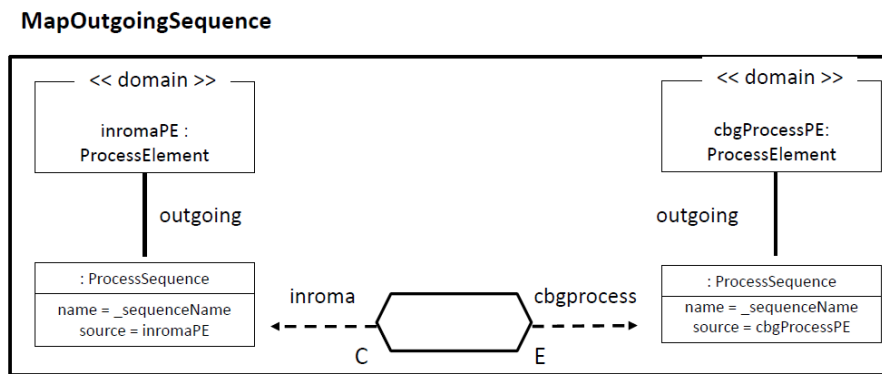


Figura A.12: Vista QVT Relations de la relación MapOutgoingSequence en la transformación M2M inroma2CBGProcess

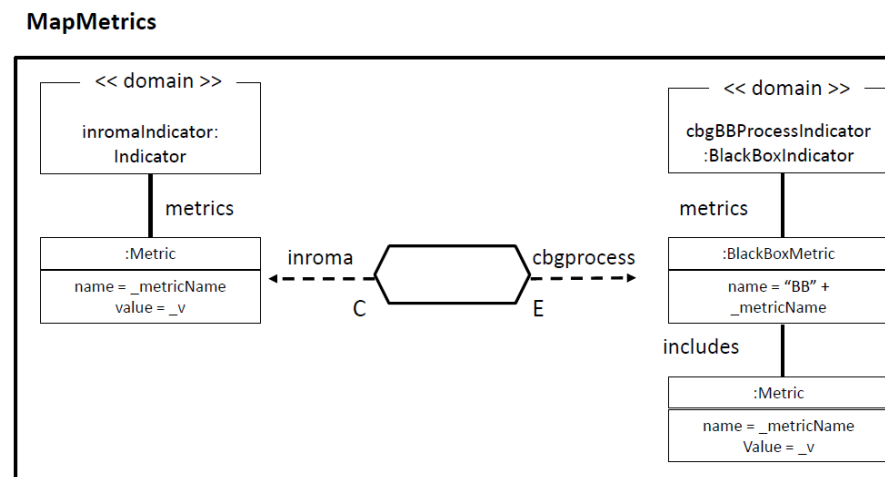


Figura A.13: Vista QVT Relations de la relación MapMetrics en la transformación M2M inroma2CBGProcess

Seguidamente se expone un fragmento del código incluido en el Add-in de la solución CBG-Tool que implementa esta transformación:

```

1 namespace CBGAddin.M2M
2 {
3     class INROMA2CBGProcessM2MHandler : CBGPROCESSHandler, IM2MTransformation
4     {
5         public INROMA2CBGProcessM2MHandler() {}
6
7         public EA.Package transform(EA.Repository repository, EA.Package
            inputModel, EA.Package outputModelView)
8         {
9             addNewRoot(repository, outputModelView);
10
11             int diagramID = getDiagramID();
12             repository.OpenDiagram(diagramID);
13
14             foreach (EA.Element e in inputModel.Elements)
15             {
16                 if (e.Stereotype.Contains(INROMAModelHandler.MMProcess))
17                     Process2Process(e, findElement(e.Name));
18             }
19
20             repository.SaveDiagram(diagramID);
21             _package.Update();
22             repository.RefreshModelView(_package.PackageID);
23
24             return _package;
25         }
26
27         public int getOutputDiagramID() { return getDiagramID(); }
28
29         void Process2Process(EA.Element inromaProcess, EA.Element cbgProcess)
30         {
31             if (cbgProcess == null)
32                 cbgProcess = addNewProcess(inromaProcess.Name);
33
34             if (cbgProcess == null)
35                 return;
36
37             MapActivities(inromaProcess, cbgProcess);
38             MapIndicators(inromaProcess, cbgProcess);
39             MapInitial(inromaProcess, cbgProcess);
40             MapFinal(inromaProcess, cbgProcess);
41             MapConditional(inromaProcess, cbgProcess);
42             MapOrganization(cbgProcess);
43         }
44
45         void MapActivities(EA.Element inromaProcess, EA.Element cbgProcess)
46         {
47             foreach (EA.Element activity in inromaProcess.Elements)
48             {
49                 if (activity.Stereotype.Contains(INROMAModelHandler.MMActivity))
50                 {
51                     EA.Element cbgActivity = findElementMapped(activity.ElementID)
52                     ;
53                     if (cbgActivity == null)
54                         cbgActivity = addNewBBActivity(cbgProcess.Elements,

```

```

        activity.Name);
54
55     if (cbgActivity != null)
56     {
57         addElement(activity.ElementID, cbgActivity.ElementID);
58         MapInputProductsOfActivity(activity, cbgActivity,
59             cbgProcess);
60         MapOutputProductsOfActivity(activity, cbgActivity,
61             cbgProcess);
62         MapInputDeliverablesOfActivity(activity, cbgActivity,
63             cbgProcess);
64         MapOutputDeliverablesOfActivity(activity, cbgActivity,
65             cbgProcess);
66         MapStakeholdersPartOfActivity(activity, cbgActivity,
67             cbgProcess);
68         MapStakeholdersRespOfActivity(activity, cbgActivity,
69             cbgProcess);
70         MapIncomingSequence(activity, cbgActivity);
71         MapOutgoingSequence(activity, cbgActivity);
72     }
73 }
74
75 void MapIndicators(EA.Element inromaProcess, EA.Element cbgProcess)
76 {
77     foreach (EA.Element inromaIndicator in inromaProcess.Elements)
78     {
79         if (inromaIndicator.Stereotype.Contains(INROMAModelHandler.
80             MMIndicator))
81         {
82             EA.Element cbgIndicator = findElementMapped(inromaIndicator.
83                 ElementID);
84             if (cbgIndicator == null)
85                 cbgIndicator = addNewBBIndicator(cbgProcess.Elements,
86                     inromaIndicator.Name);
87
88             if (cbgIndicator != null)
89             {
90                 addElement(inromaIndicator.ElementID, cbgIndicator.
91                     ElementID);
92                 MapMetrics(inromaIndicator, cbgIndicator);
93             }
94         }
95     }
96 }
97
98 void MapMetrics(EA.Element inromaIndicator, EA.Element cbgIndicator)
99 {
100     foreach (EA.Element inromaMetric in inromaIndicator.Elements)
101     {
102         if (inromaMetric.Stereotype.Contains(INROMAModelHandler.MMMetric))
103         {
104             EA.Element cbgMetric = findElementMapped(inromaMetric.
105                 ElementID);
106             if (cbgMetric == null)
107                 cbgMetric = addNewBBMetric(cbgIndicator.Elements,
108                     inromaMetric.Name);

```

```

99
100         if (cbgMetric != null)
101         {
102             addElement(inromaMetric.ElementID, cbgMetric.ElementID);
103         }
104     }
105 }
106
107
108 void MapInitial(EA.Element inromaProcess, EA.Element cbgProcess)
109 {
110     foreach (EA.Element e in inromaProcess.Elements)
111     {
112         if (e.Stereotype.Contains(INROMAModelHandler.MMInitial))
113         {
114             EA.Element cbgInitial = findElementMapped(e.ElementID);
115             if (cbgInitial == null)
116                 cbgInitial = addNewInitial(cbgProcess.Elements, e.Name);
117
118             if (cbgInitial != null)
119             {
120                 addElement(e.ElementID, cbgInitial.ElementID);
121                 MapOutgoingSequence(e, cbgInitial);
122             }
123         }
124     }
125 }
126
127 void MapFinal(EA.Element inromaProcess, EA.Element cbgProcess)
128 {
129     foreach (EA.Element e in inromaProcess.Elements)
130     {
131         if (e.Stereotype.Contains(INROMAModelHandler.MMFinal))
132         {
133             EA.Element cbgFinal = findElementMapped(e.ElementID);
134             if (cbgFinal == null)
135                 cbgFinal = addNewFinal(cbgProcess.Elements, e.Name);
136
137             if (cbgFinal != null)
138             {
139                 addElement(e.ElementID, cbgFinal.ElementID);
140                 MapIncomingSequence(e, cbgFinal);
141             }
142         }
143     }
144 }
145
146 void MapConditional(EA.Element inromaProcess, EA.Element cbgProcess)
147 {
148     foreach (EA.Element e in inromaProcess.Elements)
149     {
150         if (e.Stereotype.Contains(INROMAModelHandler.MMConditional))
151         {
152             EA.Element cbgConditional = findElementMapped(e.ElementID);
153             if (cbgConditional == null)
154                 cbgConditional = addNewConditional(cbgProcess.Elements, e.
155                     Name);

```

```

156         if (cbgConditional != null)
157         {
158             addElement(e.ElementID, cbgConditional.ElementID);
159             MapIncomingSequence(e, cbgConditional);
160             MapOutgoingSequence(e, cbgConditional);
161         }
162     }
163 }
164
165
166 void MapIncomingSequence(EA.Element inromaElement, EA.Element cbgElement)
167 {
168     foreach (EA.Connector c in inromaElement.Connectors)
169     {
170         if (c.Stereotype.Contains(INROMAModelHandler.MMProcessSequence) ==
171             false)
172             continue;
173
174         if (c.SupplierID != inromaElement.ElementID) // not incoming
175             continue;
176
177         EA.Element source = findElementMapped(c.ClientID);
178         EA.Element target = cbgElement;
179
180         if (source != null && target != null)
181             addNewProcessSequence(source, target, c.Name);
182     }
183
184 void MapOutgoingSequence(EA.Element inromaElement, EA.Element cbgElement)
185 {
186     foreach (EA.Connector c in inromaElement.Connectors)
187     {
188         if (c.Stereotype.Contains(INROMAModelHandler.MMProcessSequence) ==
189             false)
190             continue;
191
192         if (c.ClientID != inromaElement.ElementID) // not outgoing
193             continue;
194
195         EA.Element source = cbgElement;
196         EA.Element target = findElementMapped(c.SupplierID);
197
198         if (source != null && target != null)
199             addNewProcessSequence(source, target, c.Name);
200     }
201
202 void MapInputProductsOfActivity(EA.Element inromaActivity, EA.Element
203     cbgTask, EA.Element cbgProcess)
204 {
205     foreach (EA.Connector c in inromaActivity.Connectors)
206     {
207         int elementID = c.SupplierID == inromaActivity.ElementID ? c.
208             ClientID : c.SupplierID;
209         EA.Element e = _repository.GetElementByID(elementID);
210         if (e != null && c.Stereotype.Contains(INROMAModelHandler.MMEntry)
211             )

```



```

209         {
210             if (e.Stereotype.Contains(INROMAModelHandler.MMProduct) ==
                false)
211                 continue;
212             EA.Element product = findElementMapped(e.ElementID);
213             if (product == null)
214                 product = addNewBBProduct(cbgProcess.Elements, e.Name);
215
216             if (product != null)
217             {
218                 addElement(e.ElementID, product.ElementID);
219                 addNewEntry(cbgTask, product, "");
220             }
221         }
222     }
223 }
224
225 void MapOutputProductsOfActivity(EA.Element inromaActivity, EA.Element
    cbgTask, EA.Element cbgProcess)
226 {
227     foreach (EA.Connector c in inromaActivity.Connectors)
228     {
229         int elementID = c.SupplierID == inromaActivity.ElementID ? c.
            ClientID : c.SupplierID;
230         EA.Element e = _repository.GetElementByID(elementID);
231         if (e != null && c.Stereotype.Contains(INROMAModelHandler.MMExit))
232         {
233             if (e.Stereotype.Contains(INROMAModelHandler.MMProduct) ==
                false)
234                 continue;
235             EA.Element product = findElementMapped(e.ElementID);
236             if (product == null)
237                 product = addNewBBProduct(cbgProcess.Elements, e.Name);
238
239             if (product != null)
240             {
241                 addElement(e.ElementID, product.ElementID);
242                 addNewExit(cbgTask, product, "");
243             }
244         }
245     }
246 }
247
248 void MapInputDeliverablesOfActivity(EA.Element inromaActivity, EA.Element
    cbgTask, EA.Element cbgProcess)
249 {
250     foreach (EA.Connector c in inromaActivity.Connectors)
251     {
252         int elementID = c.SupplierID == inromaActivity.ElementID ? c.
            ClientID : c.SupplierID;
253         EA.Element e = _repository.GetElementByID(elementID);
254         if (e != null && c.Stereotype.Contains(INROMAModelHandler.MMEntry)
            )
255         {
256             if (e.Stereotype.Contains(INROMAModelHandler.MMDeliverable) ==
                false)
257                 continue;
258             EA.Element deliverable = findElementMapped(e.ElementID);

```

```

259         if (deliverable == null)
260             deliverable = addNewDeliverable(cbgProcess.Elements, e.
                Name);
261
262         if (deliverable != null)
263         {
264             addElement(e.ElementID, deliverable.ElementID);
265             addNewEntry(cbgTask, deliverable, "");
266         }
267     }
268 }
269 }
270
271 void MapOutputDeliverablesOfActivity(EA.Element inromaActivity, EA.Element
    cbgTask, EA.Element cbgProcess)
272 {
273     foreach (EA.Connector c in inromaActivity.Connectors)
274     {
275         int elementID = c.SupplierID == inromaActivity.ElementID ? c.
            ClientID : c.SupplierID;
276         EA.Element e = _repository.GetElementByID(elementID);
277         if (e != null && c.Stereotype.Contains(INROMAModelHandler.MMExit))
278         {
279             if (e.Stereotype.Contains(INROMAModelHandler.MMDeliverable) ==
                false)
280                 continue;
281             EA.Element deliverable = findElementMapped(e.ElementID);
282             if (deliverable == null)
283                 deliverable = addNewDeliverable(cbgProcess.Elements, e.
                    Name);
284
285             if (deliverable != null)
286             {
287                 addElement(e.ElementID, deliverable.ElementID);
288                 addNewExit(cbgTask, deliverable, "");
289             }
290         }
291     }
292 }
293
294 void MapStakeholdersPartOfActivity(EA.Element inromaActivity, EA.Element
    cbgTask, EA.Element cbgProcess)
295 {
296     foreach (EA.Connector c in inromaActivity.Connectors)
297     {
298         int elementID = c.SupplierID == inromaActivity.ElementID ? c.
            ClientID : c.SupplierID;
299         EA.Element e = _repository.GetElementByID(elementID);
300
301         if (e != null && c.Stereotype.Contains(INROMAModelHandler.
            MMParticipes))
302         {
303             if (e.Stereotype.Contains(INROMAModelHandler.MMStakeholder) ==
                false)
304                 continue;
305             EA.Element resourceRole = findElementMapped(e.ElementID);
306             if (resourceRole == null)
307                 resourceRole = addNewStakeholder(cbgProcess.Elements, e.

```

```

308         Name);
309     if (resourceRole != null)
310     {
311         addElement(e.ElementID, resourceRole.ElementID);
312         addNewParticipes(cbgTask, resourceRole, "");
313     }
314 }
315 }
316 }
317
318 void MapStakeholdersRespOfActivity(EA.Element inromaActivity, EA.Element
    cbgTask, EA.Element cbgProcess)
319 {
320     foreach (EA.Connector c in inromaActivity.Connectors)
321     {
322         int elementID = c.SupplierID == inromaActivity.ElementID ? c.
            ClientID : c.SupplierID;
323         EA.Element e = _repository.GetElementByID(elementID);
324         if (e != null && c.Stereotype.Contains(INROMAModelHandler.
            MMIsResponsible))
325         {
326             if (e.Stereotype.Contains(INROMAModelHandler.MMStakeholder) ==
                false)
327                 continue;
328             EA.Element resourceRole = findElementMapped(e.ElementID);
329             if (resourceRole == null)
330                 resourceRole = addNewStakeholder(cbgProcess.Elements, e.
                    Name);
331
332             if (resourceRole != null)
333             {
334                 addElement(e.ElementID, resourceRole.ElementID);
335                 addNewIsResponsible(cbgTask, resourceRole, "");
336             }
337         }
338     }
339 }
340
341 void MapOrganization(EA.Element cbgProcess)
342 {
343     addNewOrganization(cbgProcess.Elements, "MyOrganization");
344 }
345
346 }
347 }
348 }

```

Algoritmo A.1: Implementación de la transformación INROMA2CBGPROCESS en C#

Anexo B

Transformaciones M2M entre CBGProcess y CBGCase

En este anexo se presentan las relaciones para acometer la transformación bidireccional entre CBGProcess y CBGCase. Para ello, haremos uso de la sintaxis concreta gráfica de QVT Relations, tal y como fueron descritas en el capítulo 6.

La tabla B.1 expone las correspondencias entre las relaciones y los diagramas de transformación QVT que se muestran a continuación.

Relación	Sintaxis Gráfica QVT Relacional
Process2Case	Figura B.1
MapOrganizations	Figura B.2
MapActivities	Figura B.3
MapInitial	Figura B.4
MapFinal	Figura B.5
MapConditional	Figura B.6
MapStakeholders	Figura B.7
MapInputProductsOfActivity	Figura B.8
MapOutputProductsOfActivity	Figura B.9
MapIncludedInputProducts	Figura B.10
MapIncludedInputProducts	Figura B.10
MapIncludedOutputProducts	Figura B.11
MapIncludedActivitiesOfActivity	Figura B.12
MapStakeholdersPartOfActivity	Figura B.13
MapStakeholdersRespOfActivity	Figura B.14
MapIncomingSequence	Figura B.15
MapOutgoingSequence	Figura B.16

Tabla B.1: Relaciones para acometer la transformación CBGProcess2CBGCase

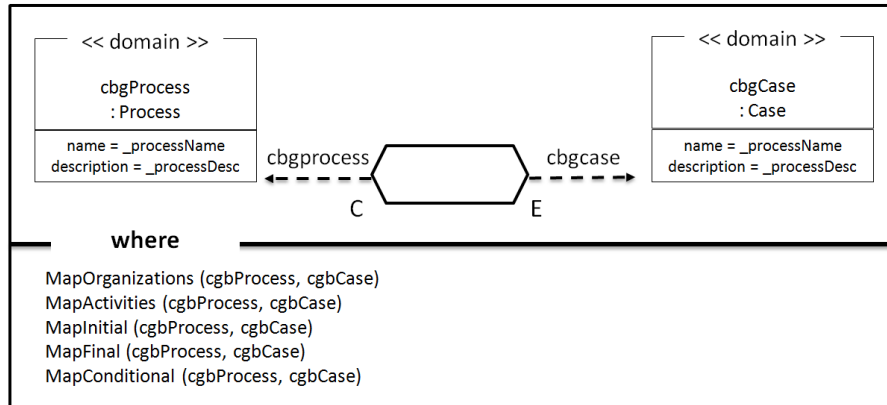
Process2Case

Figura B.1: Vista QVT Relations de la relación Process2Case en la transformación M2M CBG-Process2CBGCase

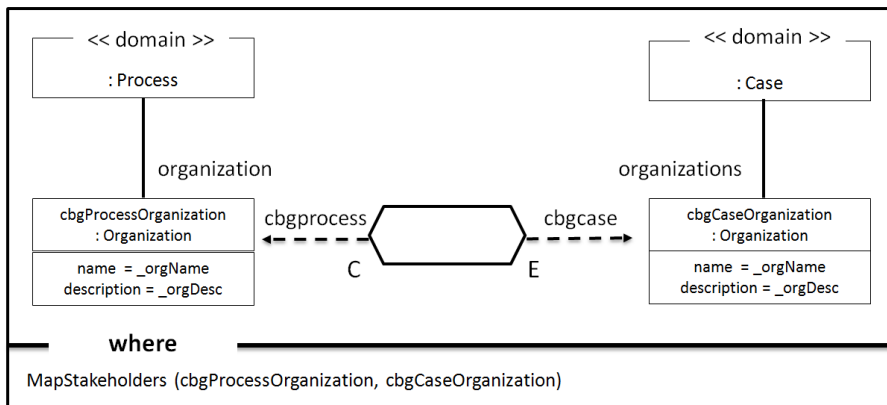
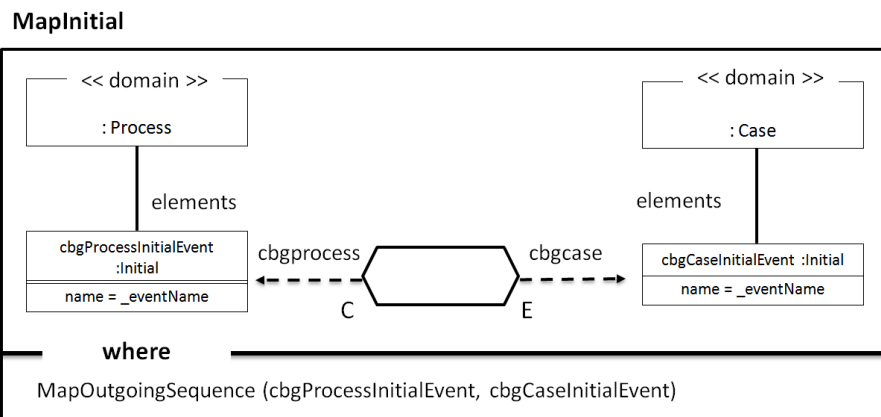
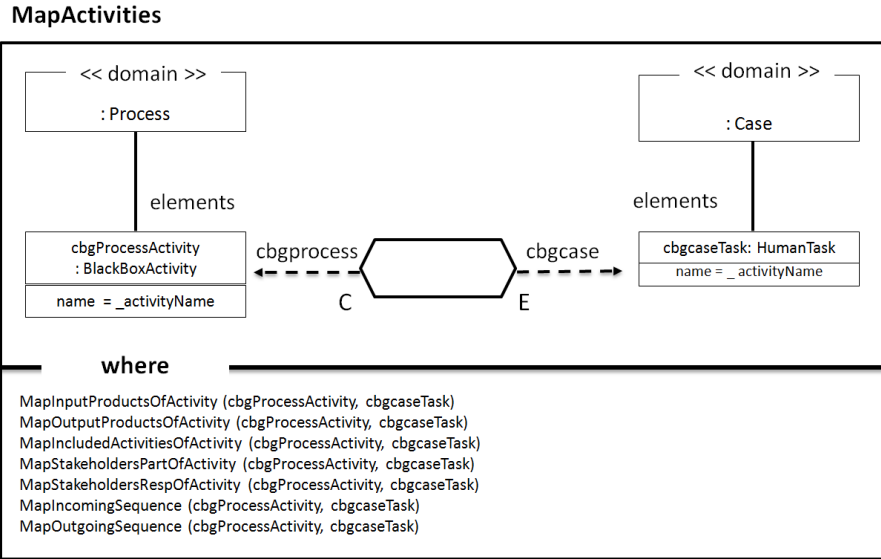
MapOrganizations

Figura B.2: Vista QVT Relations de la relación MapOrganizations en la transformación M2M CBGProcess2CBGCase



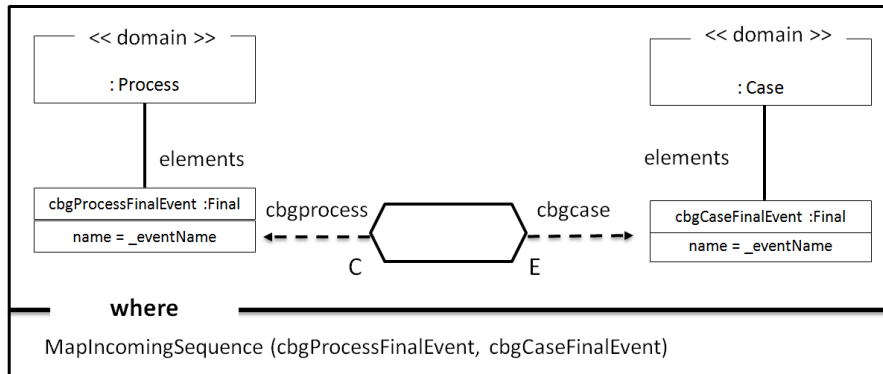
MapFinal

Figura B.5: Vista QVT Relations de la relación MapFinal en la transformación M2M CBGProcess2CBGCase

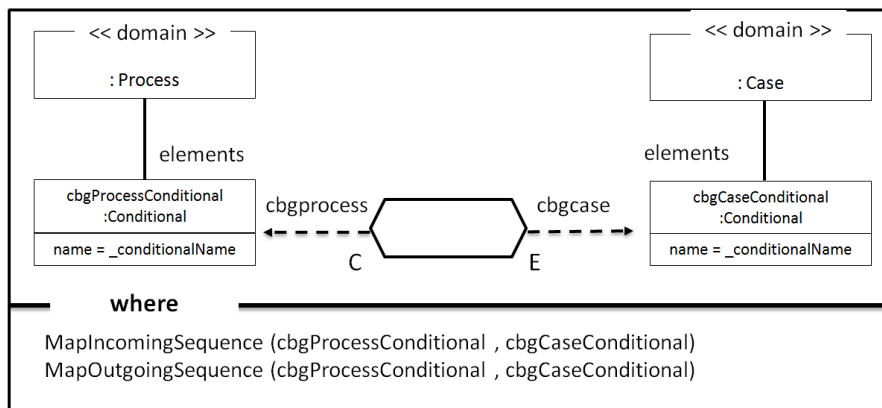
MapConditional

Figura B.6: Vista QVT Relations de la relación MapConditional en la transformación M2M CBGProcess2CBGCase

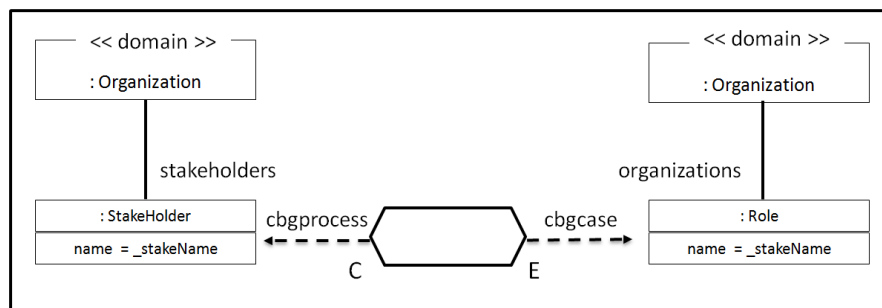
MapStakeholders

Figura B.7: Vista QVT Relations de la relación MapStakeholders en la transformación M2M CBGProcess2CBGCase

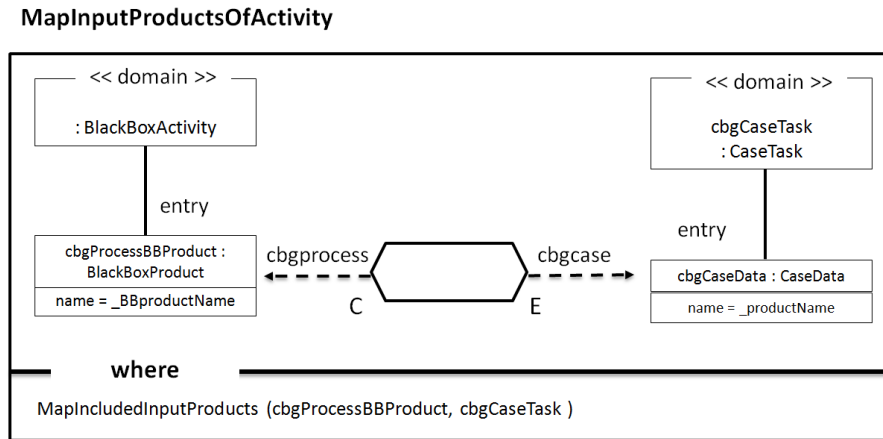


Figura B.8: Vista QVT Relations de la relación MapInputProductsOfActivity en la transformación M2M CBGProcess2CBGCase

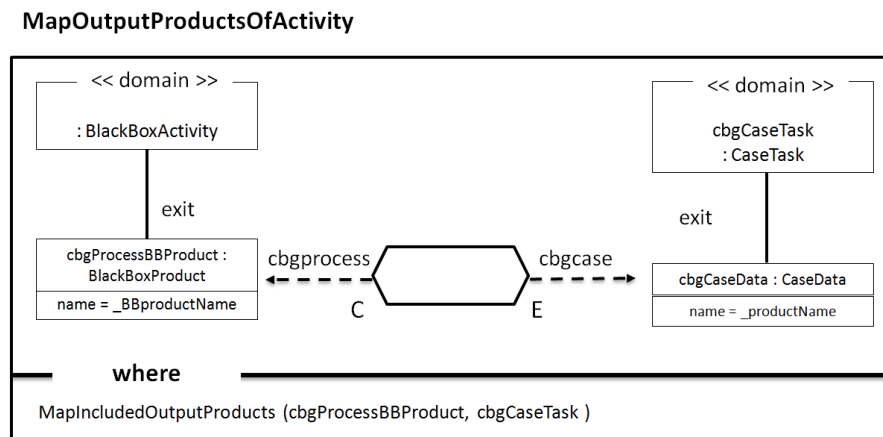


Figura B.9: Vista QVT Relations de la relación MapOutputProductsOfActivity en la transformación M2M CBGProcess2CBGCase

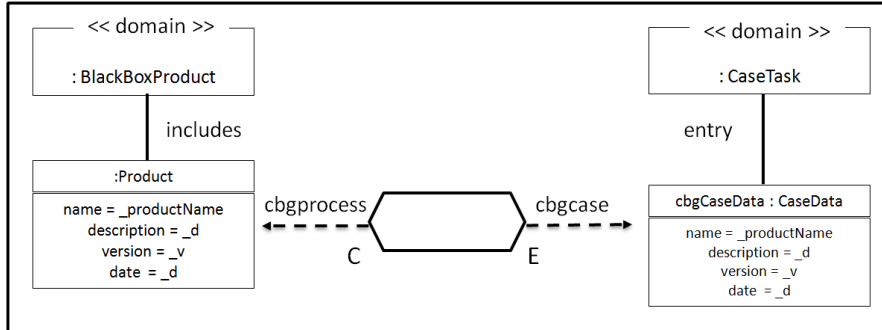
MapIncludedInputProducts

Figura B.10: Vista QVT Relations de la relación MapIncludedInputProducts en la transformación M2M CBGProcess2CBGCase

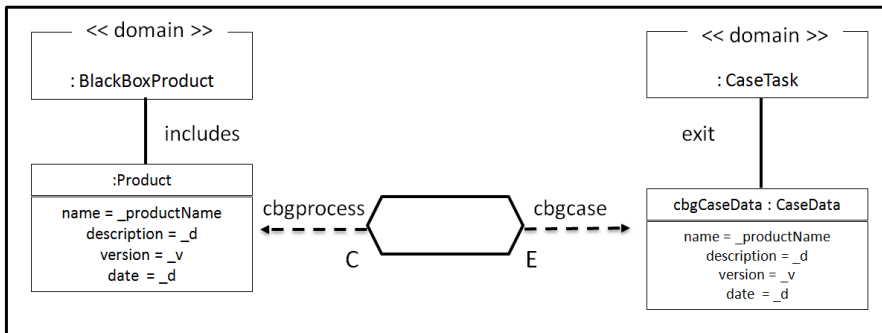
MapIncludedOutputProducts

Figura B.11: Vista QVT Relations de la relación MapIncludedOutputProducts en la transformación M2M CBGProcess2CBGCase

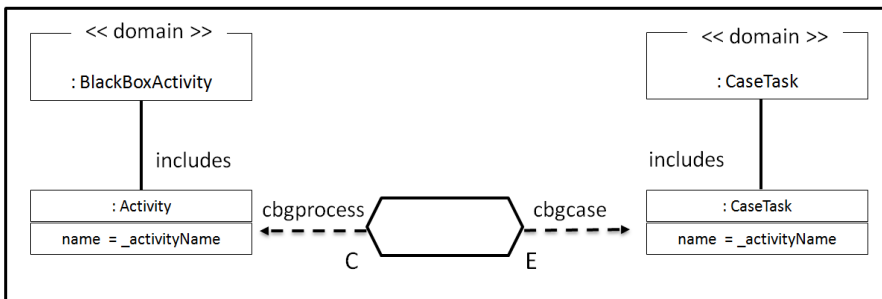
MapIncludedActivitiesOfActivity

Figura B.12: Vista QVT Relations de la relación MapIncludedActivitiesOfActivity en la transformación M2M CBGProcess2CBGCase

MapStakeholdersPartOfActivity

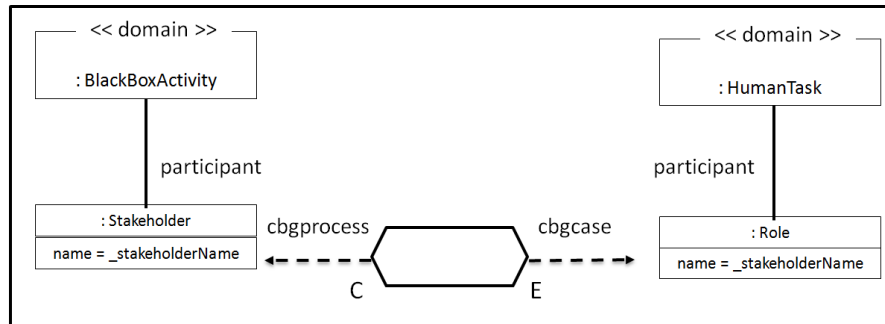


Figura B.13: Vista QVT Relations de la relación MapStakeholdersPartOfActivity en la transformación M2M CBGProcess2CBGCase

MapStakeholdersRespOfActivity

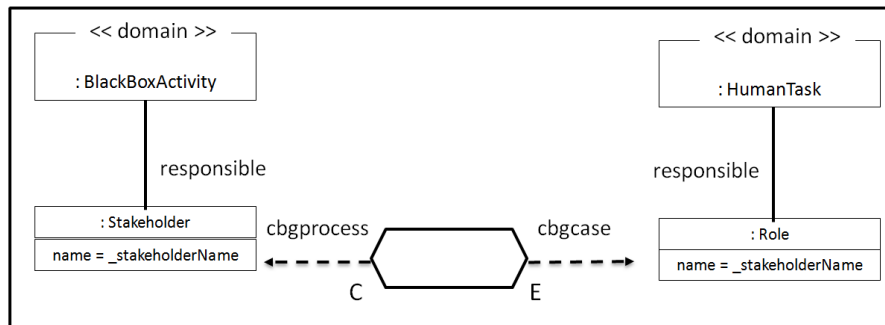


Figura B.14: Vista QVT Relations de la relación MapStakeholdersRespOfActivity en la transformación M2M CBGProcess2CBGCase

MapIncomingSequence

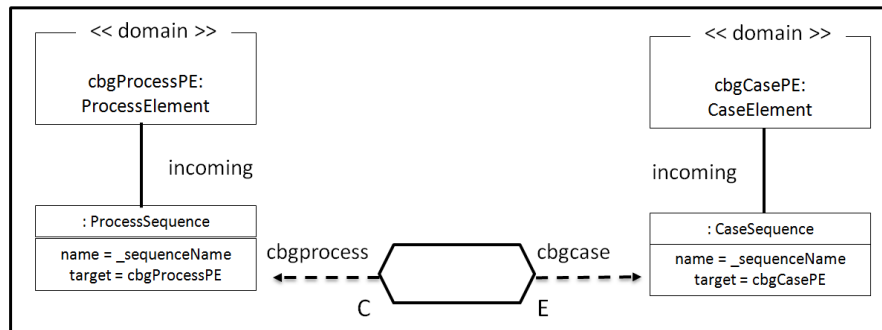


Figura B.15: Vista QVT Relations de la relación MapIncomingSequence en la transformación M2M CBGProcess2CBGCase

MapOutgoingSequence

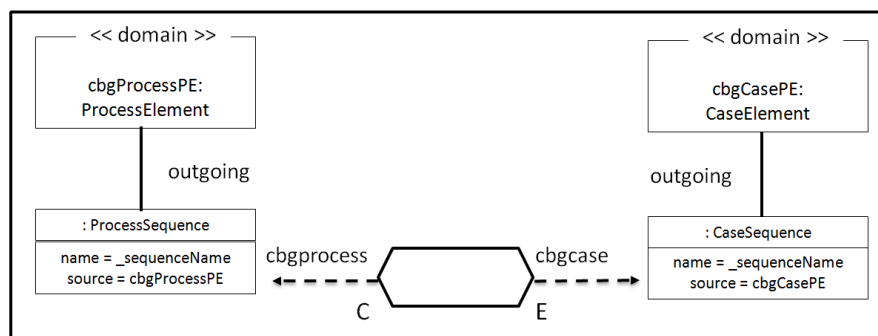


Figura B.16: Vista QVT Relations de la relación MapOutgoingSequence en la transformación M2M CBGProcess2CBGCase

Seguidamente se expone un fragmento del código incluido en el Add-in de la solución CBG-Tool que implementa esta transformación:

```

1 namespace CBGAddin.M2M
2 {
3     class CGProcess2CBGCaseM2MHandler : CBGCASEHandler, IM2MTransformation
4     {
5
6         public const string TAG_VALUE_NAME_ISCOLLABORATIVE = "isCollaborative";
7         public const string TAG_VALUE_INDIVIDUAL = "INDIVIDUAL";
8         public const string TAG_VALUE_COLLABORATIVE = "COLLABORATIVE";
9
10        public CGProcess2CBGCaseM2MHandler() {}
11
12        public EA.Package transform(EA.Repository repository, EA.Package
            inputModel, EA.Package outputModelView)
13        {
14            addNewRoot(repository, outputModelView);
15
16            int diagramID = getDiagramID();
17            repository.OpenDiagram(diagramID);
18
19            foreach (EA.Element e in inputModel.Elements)
20            {
21                if (e.Stereotype.Contains(CBGPROCESSHandler.MMProcess))
22                    Process2Case(e, findElement(e.Name));
23            }
24
25            repository.SaveDiagram(diagramID);
26            _package.Update();
27            repository.RefreshModelView(_package.PackageID);
28
29            return _package;
30        }
31
32        public int getOutputDiagramID() { return getDiagramID(); }
33
34        void Process2Case(EA.Element cbgProcess, EA.Element cbgCase)
35        {
36            if (cbgCase == null)
37                cbgCase = addNewCase(cbgProcess.Name);
38
39            if (cbgCase == null)
40                return;
41
42            MapOrganizations(cbgProcess, cbgCase);
43            MapActivities(cbgProcess, cbgCase);
44            MapInitial(cbgProcess, cbgCase);
45            MapFinal(cbgProcess, cbgCase);
46            MapConditional(cbgProcess, cbgCase);
47        }
48
49        void MapOrganizations(EA.Element cbgProcess, EA.Element cbgCase)
50        {
51            foreach (EA.Element processOrganization in cbgProcess.Elements)
52            {
53                if (processOrganization.Stereotype.Contains(CBGPROCESSHandler.
                    MMOrganization))

```

```

54         {
55             EA.Element caseOrganization = findElementMapped(
                    processOrganization.ElementID);
56             if (caseOrganization == null)
57                 caseOrganization = addNewOrganization(cbgCase.Elements,
                    processOrganization.Name);
58
59             if (caseOrganization != null)
60             {
61                 addElement(processOrganization.ElementID, caseOrganization
                    .ElementID);
62                 MapStakeholders(processOrganization, caseOrganization,
                    cbgCase);
63             }
64         }
65     }
66 }
67
68 void MapStakeholders(EA.Element processOrganization, EA.Element
    caseOrganization, EA.Element cbgCase)
69 {
70     foreach (EA.Connector c in processOrganization.Connectors)
71     {
72         int elementID = c.SupplierID == processOrganization.ElementID ? c.
            ClientID : c.SupplierID;
73         EA.Element e = _repository.GetElementByID(elementID);
74
75         if (e.Stereotype.Contains(CBGPROCESSHandler.MMStakeholder) ==
            false)
76             continue;
77         EA.Element caseRole = findElementMapped(e.ElementID);
78         if (caseRole == null)
79             caseRole = addNewRole(cbgCase.Elements, e.Name);
80
81         if (caseRole != null)
82         {
83             addElement(e.ElementID, caseRole.ElementID);
84             addNewParticipes(caseOrganization, caseRole, "");
85         }
86     }
87 }
88
89 void MapActivities(EA.Element cbgProcess, EA.Element cbgCase)
90 {
91     foreach (EA.Element activity in cbgProcess.Elements)
92     {
93         if (activity.Stereotype.Contains(CBGPROCESSHandler.MMActivity))
94         {
95             EA.Element cbgCaseTask = findElementMapped(activity.ElementID)
                ;
96
97             // individual or collaborative
98             bool isCollaborative = isCBGProcessElementCollaborative(
                cbgProcess);
99
100             if (cbgCaseTask == null)
101                 cbgCaseTask = addNewHumanCaseTask(cbgCase.Elements,
                    activity.Name, isCollaborative); // default all Human

```

```

102         if (cbgCaseTask != null)
103         {
104             addElement(activity.ElementID, cbgCaseTask.ElementID);
105             MapInputProductsOfActivity(activity, cbgCaseTask, cbgCase)
106             ;
107             MapOutputProductsOfActivity(activity, cbgCaseTask, cbgCase
108             );
109             MapIncludedActivitiesOfActivity(activity, cbgCaseTask,
110             cbgCase);
111             MapStakeholdersPartOfActivity(activity, cbgCaseTask,
112             cbgCase);
113             MapStakeholdersRespOfActivity(activity, cbgCaseTask,
114             cbgCase);
115             MapIncomingSequence(activity, cbgCaseTask);
116             MapOutgoingSequence(activity, cbgCaseTask);
117         }
118     }
119 }
120
121 void MapInitial(EA.Element cbgProcess, EA.Element cbgCase)
122 {
123     foreach (EA.Element e in cbgProcess.Elements)
124     {
125         if (e.Stereotype.Contains(CBGPROCESSHandler.MMInitial))
126         {
127             EA.Element cbgInitial = findElementMapped(e.ElementID);
128             if (cbgInitial == null)
129                 cbgInitial = addNewInitial(cbgCase.Elements, e.Name);
130
131             if (cbgInitial != null)
132             {
133                 addElement(e.ElementID, cbgInitial.ElementID);
134                 MapOutgoingSequence(e, cbgInitial);
135             }
136         }
137     }
138 }
139
140 void MapFinal(EA.Element cbgProcess, EA.Element cbgCase)
141 {
142     foreach (EA.Element e in cbgProcess.Elements)
143     {
144         if (e.Stereotype.Contains(CBGPROCESSHandler.MMFinal))
145         {
146             EA.Element cbgFinal = findElementMapped(e.ElementID);
147             if (cbgFinal == null)
148                 cbgFinal = addNewFinal(cbgCase.Elements, e.Name);
149
150             if (cbgFinal != null)
151             {
152                 addElement(e.ElementID, cbgFinal.ElementID);
153                 MapIncomingSequence(e, cbgFinal);
154             }
155         }
156     }
157 }

```

```

155 void MapConditional(EA.Element cbgProcess, EA.Element cbgCase)
156 {
157     foreach (EA.Element e in cbgProcess.Elements)
158     {
159         if (e.Stereotype.Contains(CBGPROCESSHandler.MMConditional))
160         {
161             EA.Element cbgConditional = findElementMapped(e.ElementID);
162             if (cbgConditional == null)
163                 cbgConditional = addNewConditional(cbgCase.Elements, e.
164                     Name);
165
166             if (cbgConditional != null)
167             {
168                 addElement(e.ElementID, cbgConditional.ElementID);
169                 MapIncomingSequence(e, cbgConditional);
170                 MapOutgoingSequence(e, cbgConditional);
171             }
172         }
173     }
174 }
175
176 void MapIncomingSequence(EA.Element inromaElement, EA.Element cbgElement)
177 {
178     foreach (EA.Connector c in inromaElement.Connectors)
179     {
180         if (c.Stereotype.Contains(CBGPROCESSHandler.MMProcessSequence) ==
181             false)
182             continue;
183
184         if (c.SupplierID != inromaElement.ElementID) // not incoming
185             continue;
186
187         EA.Element source = findElementMapped(c.ClientID);
188         EA.Element target = cbgElement;
189
190         if (source != null && target != null)
191             addNewCaseSequence(source, target, c.Name);
192     }
193
194 void MapOutgoingSequence(EA.Element inromaElement, EA.Element cbgElement)
195 {
196     foreach (EA.Connector c in inromaElement.Connectors)
197     {
198         if (c.Stereotype.Contains(CBGPROCESSHandler.MMProcessSequence) ==
199             false)
200             continue;
201
202         if (c.ClientID != inromaElement.ElementID) // not outgoing
203             continue;
204
205         EA.Element source = cbgElement;
206         EA.Element target = findElementMapped(c.SupplierID);
207
208         if (source != null && target != null)
209             addNewMandatoryCaseSequence(source, target, c.Name); // by
210             default all mandatory

```

```

209     }
210 }
211
212 void MapInputProductsOfActivity(EA.Element inromaActivity, EA.Element
    cbgTask, EA.Element cbgCase)
213 {
214     foreach (EA.Connector c in inromaActivity.Connectors)
215     {
216         int elementID = c.SupplierID == inromaActivity.ElementID ? c.
            ClientID : c.SupplierID;
217         EA.Element e = _repository.GetElementByID(elementID);
218         if (e != null && c.Stereotype.Contains(CBGPROCESSHandler.MMEntry))
219         {
220             if (e.Stereotype.Contains(CBGPROCESSHandler.MMProduct) ==
                false)
221                 continue;
222             EA.Element caseData = findElementMapped(e.ElementID);
223
224             bool isCollaborative = isCBGProcessElementCollaborative(e);
225             if (caseData == null)
226                 caseData = addNewCaseData(cbgCase.Elements, e.Name,
                    isCollaborative);
227
228             if (caseData != null)
229             {
230                 addElement(e.ElementID, caseData.ElementID);
231                 addNewEntry(cbgTask, caseData, "");
232                 MapIncludedInputProducts(e, caseData, cbgTask);
233             }
234         }
235     }
236 }
237
238 void MapOutputProductsOfActivity(EA.Element inromaActivity, EA.Element
    cbgTask, EA.Element cbgCase)
239 {
240     foreach (EA.Connector c in inromaActivity.Connectors)
241     {
242         int elementID = c.SupplierID == inromaActivity.ElementID ? c.
            ClientID : c.SupplierID;
243         EA.Element e = _repository.GetElementByID(elementID);
244         if (e != null && c.Stereotype.Contains(CBGPROCESSHandler.MMExit))
245         {
246             if (e.Stereotype.Contains(CBGPROCESSHandler.MMProduct) ==
                false)
247                 continue;
248             EA.Element caseData = findElementMapped(e.ElementID);
249             bool isCollaborative = isCBGProcessElementCollaborative(e);
250             if (caseData == null)
251                 caseData = addNewCaseData(cbgCase.Elements, e.Name,
                    isCollaborative);
252
253             if (caseData != null)
254             {
255                 addElement(e.ElementID, caseData.ElementID);
256                 addNewExit(cbgTask, caseData, "");
257                 MapIncludedOutputProducts(e, caseData, cbgTask);
258             }
259         }
260     }
261 }

```



```

259         }
260     }
261 }
262 }
263
264 void MapStakeholdersPartOfActivity(EA.Element inromaActivity, EA.Element
    cbgTask, EA.Element cbgCase)
265 {
266     foreach (EA.Connector c in inromaActivity.Connectors)
267     {
268         int elementID = c.SupplierID == inromaActivity.ElementID ? c.
            ClientID : c.SupplierID;
269         EA.Element e = _repository.GetElementByID(elementID);
270
271         if (e != null && c.Stereotype.Contains(CBGPROCESSHandler.
            MMParticipes))
272         {
273             if (e.Stereotype.Contains(CBGPROCESSHandler.MMStakeholder) ==
                false)
274                 continue;
275             EA.Element resourceRole = findElementMapped(e.ElementID);
276             if (resourceRole == null)
277                 resourceRole = addNewRole(cbgCase.Elements, e.Name);
278
279             if (resourceRole != null)
280             {
281                 addElement(e.ElementID, resourceRole.ElementID);
282                 addNewParticipes(cbgTask, resourceRole, "");
283             }
284         }
285     }
286 }
287
288 void MapStakeholdersRespOfActivity(EA.Element inromaActivity, EA.Element
    cbgTask, EA.Element cbgCase)
289 {
290     foreach (EA.Connector c in inromaActivity.Connectors)
291     {
292         int elementID = c.SupplierID == inromaActivity.ElementID ? c.
            ClientID : c.SupplierID;
293         EA.Element e = _repository.GetElementByID(elementID);
294
295         if (e != null && c.Stereotype.Contains(CBGPROCESSHandler.
            MMIIsResponsible))
296         {
297             if (e.Stereotype.Contains(CBGPROCESSHandler.MMStakeholder) ==
                false)
298                 continue;
299             EA.Element resourceRole = findElementMapped(e.ElementID);
300             if (resourceRole == null)
301                 resourceRole = addNewRole(cbgCase.Elements, e.Name);
302
303             if (resourceRole != null)
304             {
305                 addElement(e.ElementID, resourceRole.ElementID);
306                 addNewIsResponsible(cbgTask, resourceRole, "");
307             }
308         }

```

```

309     }
310 }
311
312 void MapIncludedActivitiesOfActivity(EA.Element processActivity, EA.
    Element caseTask, EA.Element cbgCase)
313 {
314     // individual or collaborative
315     bool isCollaborative = isCBGProcessElementCollaborative(
        processActivity);
316
317     foreach (EA.Connector c in processActivity.Connectors)
318     {
319         int elementID = c.SupplierID == processActivity.ElementID ? c.
            ClientID : c.SupplierID;
320         EA.Element e = _repository.GetElementByID(elementID);
321
322         if (e.Stereotype.Contains(CBGPROCESSHandler.MMActivity) == false)
323             continue;
324         EA.Element caseIncludedTask = findElementMapped(e.ElementID);
325         if (caseIncludedTask == null)
326             caseIncludedTask = addNewHumanCaseTask(cbgCase.Elements, e.
                Name, isCollaborative);
327
328         if (caseIncludedTask != null)
329         {
330             addElement(e.ElementID, caseIncludedTask.ElementID);
331             addNewIncludes(caseTask, caseIncludedTask, "");
332         }
333     }
334 }
335
336 void MapIncludedInputProducts(EA.Element processProduct, EA.Element
    cbgTask, EA.Element cbgCase)
337 {
338     foreach (EA.Connector c in processProduct.Connectors)
339     {
340         int elementID = c.SupplierID == processProduct.ElementID ? c.
            ClientID : c.SupplierID;
341         EA.Element e = _repository.GetElementByID(elementID);
342
343         if (e.Stereotype.Contains(CBGPROCESSHandler.MMProduct) == false)
344             continue;
345         EA.Element caseData = findElementMapped(e.ElementID);
346         bool isCollaborative = isCBGProcessElementCollaborative(e);
347         if (caseData == null)
348             caseData = addNewCaseData(cbgCase.Elements, e.Name,
                isCollaborative);
349
350         if (caseData != null)
351         {
352             addElement(e.ElementID, caseData.ElementID);
353             addNewEntry(cbgTask, caseData, "");
354         }
355     }
356 }
357
358 void MapIncludedOutputProducts(EA.Element processProduct, EA.Element
359

```

```

360         cbgTask, EA.Element cbgCase)
361     {
362         foreach (EA.Connector c in processProduct.Connectors)
363         {
364             int elementID = c.SupplierID == processProduct.ElementID ? c.
365                 ClientID : c.SupplierID;
366             EA.Element e = _repository.GetElementByID(elementID);
367             if (e.Stereotype.Contains(CBGPROCESSHandler.MMProduct) == false)
368                 continue;
369             EA.Element caseData = findElementMapped(e.ElementID);
370             bool isCollaborative = isCBGProcessElementCollaborative(e);
371             if (caseData == null)
372                 caseData = addNewCaseData(cbgCase.Elements, e.Name,
373                     isCollaborative);
374             if (caseData != null)
375             {
376                 addElement(e.ElementID, caseData.ElementID);
377                 addNewExit(cbgTask, caseData, "");
378             }
379         }
380     }
381     protected bool isCBGProcessElementCollaborative (EA.Element taskOrProduct)
382     {
383         try
384         {
385             // individual or collaborative
386             if (getTagValue(taskOrProduct, TAG.VALUE.NAME.ISCOLLABORATIVE) ==
387                 TAG.VALUE.INDIVIDUAL)
388                 return false;
389             else
390                 return true;
391         }
392         catch (Exception)
393         {
394             return false;
395         }
396     }
397 }
398 }

```

Algoritmo B.1: Implementación de la transformación CBGProcess2CBGCase en C#

Anexo C

Manual de usuario de CBG-Tool

Este anexo tiene como objetivo describir el uso de la herramienta CBG-Tool, a modo de un manual de usuario. Esta herramienta fue introducida en el capítulo 7, donde se explicó en detalle cómo había sido desarrollada y cómo se habían incorporado cada uno de los elementos para facilitar la creación de modelos y la ejecución de las transformaciones desde un punto de vista práctico. Sin embargo, en dicho capítulo no se hacía hincapié en el manejo de CBG-Tool, siendo este manejo el objeto de este manual de usuario.

En dicho capítulo se detalló el entorno de trabajo sobre el que se desarrolla la solución y la justificación de los motivos que nos han llevado a esa elección, por lo que en este manual de usuario se va a observar que la interfaz gráfica de CBG-Tool está integrada en el propio entorno de Enterprise Architect.

Para desarrollar el manual de usuario haremos uso de los casos de uso de la herramienta descritos en la sección 7.1.1.

C.1. Instalación del entorno de la herramienta

Como primer paso para la creación de un modelo en CBG-Tool es necesario disponer de la herramienta Enterprise Architect, junto con las siguientes extensiones:

- Perfil UML del metamodelo INROMA (disponible dentro de la herramienta MONETA).
- Perfil UML del metamodelo CBGProcess.
- Perfil UML del metamodelo CBGCase.
- Add-in CBG-Tool.

Para comprobar que los perfiles están instalados en el sistema, es preciso abrir un proyecto y, en la selección de ToolBox, pulsar *More tools* y buscar los perfiles INROMA, CBGPROCESS y CBGCASE dentro de la lista. La figura C.1 muestra la visión de la toolbox correspondiente a cada uno de ellos.

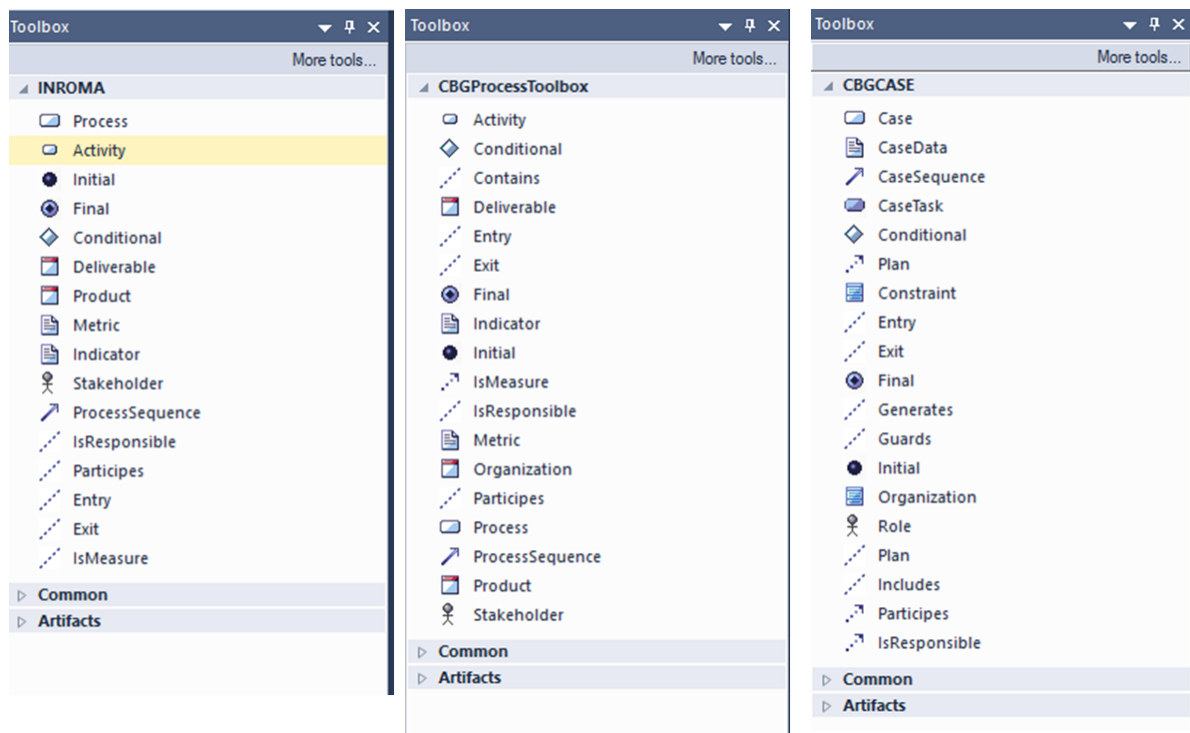


Figura C.1: Editores de INROMA, CBGProcess y CBGCase mediante perfiles UML desplegados en Enterprise Architect

Para comprobar que el Add-in está bien instalado, basta con pulsar con el botón derecho del ratón sobre un diagrama y en Extensions, encontrar el submenú CBG-Tool, como se muestra en la figura C.2.

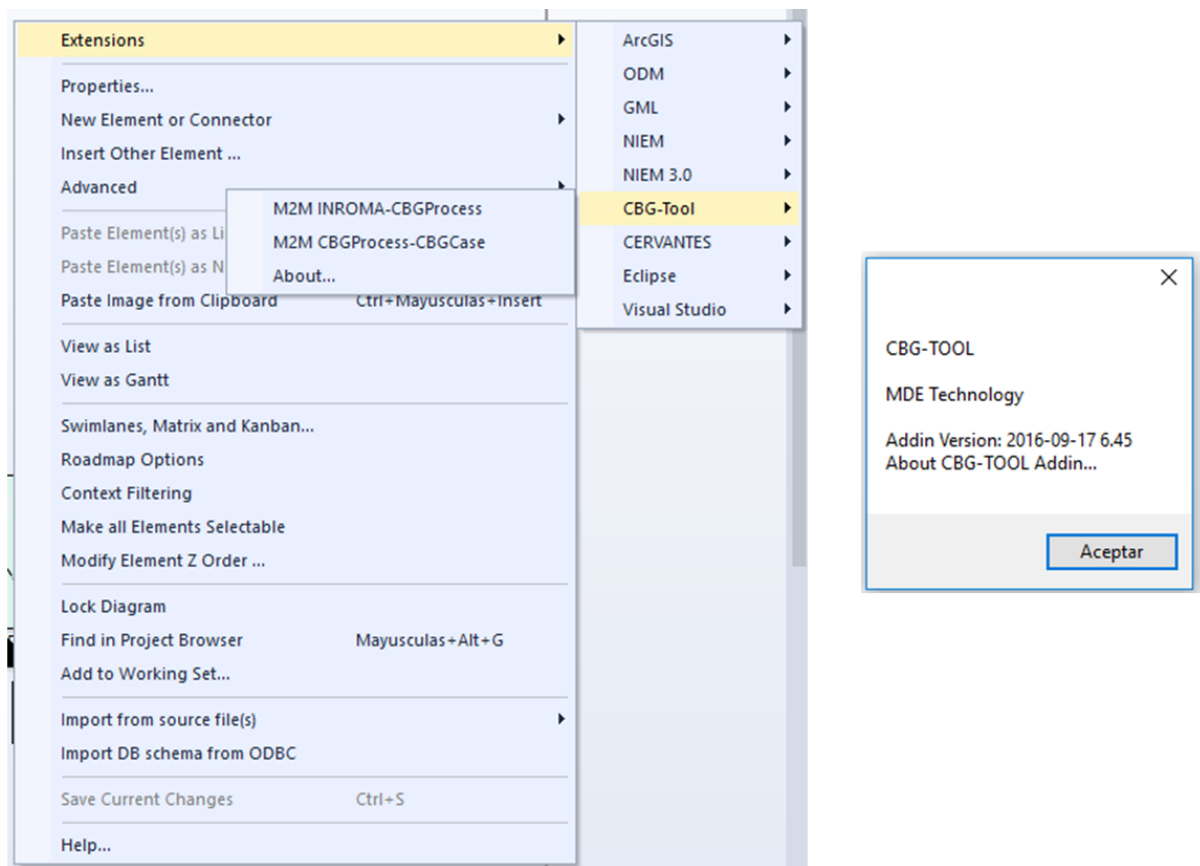


Figura C.2: Addin de CBGTool desplegado en Enterprise Architect mediante un menu contextual

C.2. Crear la vista individual del modelo de proceso colaborativo

La solución CBG-Tool parte de la descripción de un modelo de proceso conforme a INROMA, para lo cual es preciso crearlo atendiendo al perfil disponible en la herramienta MONETA. En este sentido el lector puede acudir al manual de usuario disponible en la tesis doctoral [García-Borgoñón, 2016] en la que MONETA fue desarrollada.

Una vez que tenemos la vista del diagrama INROMA del modelo de proceso, a través del menú contextual (botón derecho del ratón), podremos ir a la extensión CBG-Tool y pulsar en *M2M INROMA-CBGProcess*, tal y como se muestra en la figura C.3.

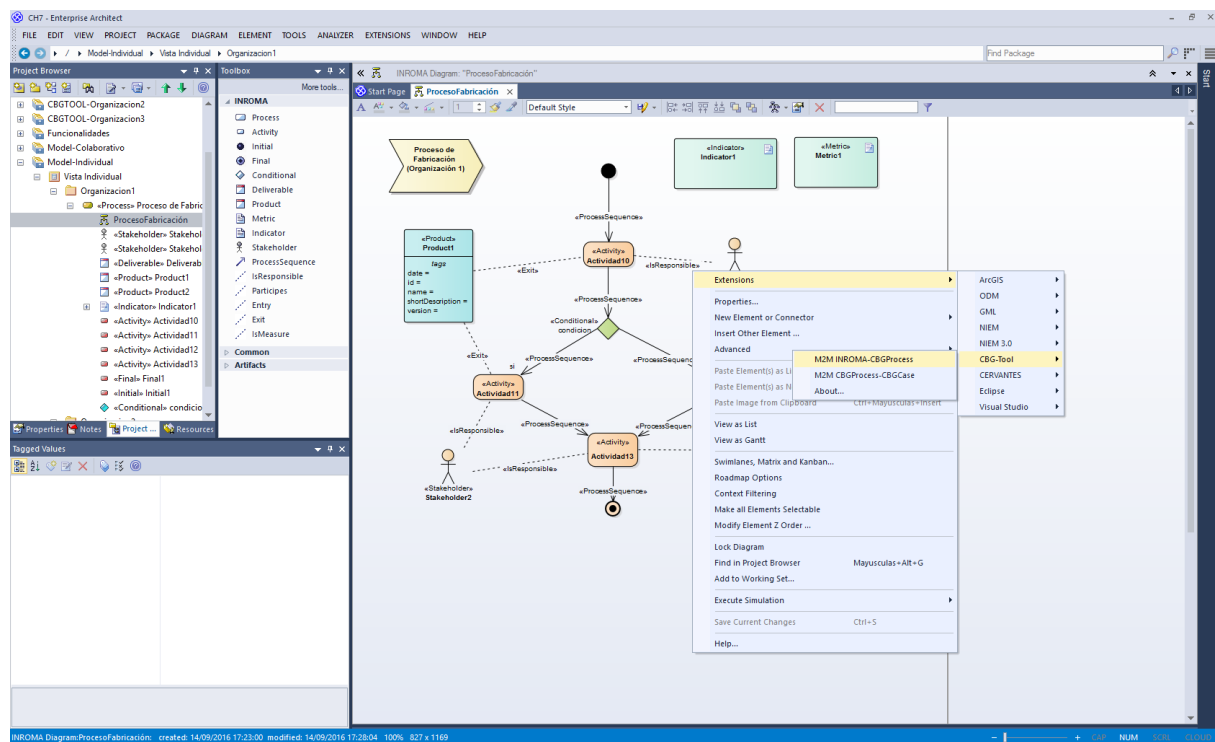


Figura C.3: Uso de CBGTool para transformar el modelo de proceso INROMA a CBGProcess

De forma automática la solución creará un nuevo modelo conforme a CBGProcess con una vista de diagrama en la que el ingeniero de procesos podrá crear la vista individual del proceso colaborativo, como expone la figura C.4.

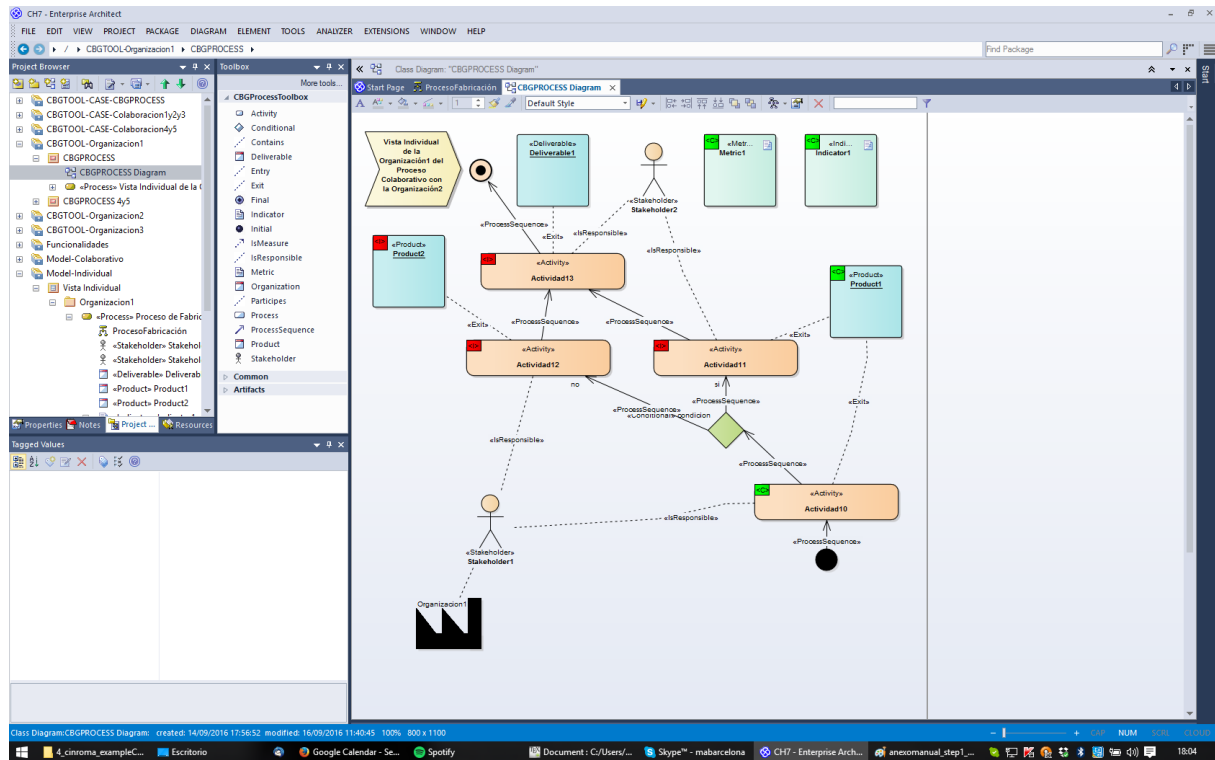


Figura C.4: Uso de CBGTool para crear la vista individual del proceso colaborativo

C.3. Crear el modelo de proceso colaborativo

A partir de las vistas individuales de los procesos en colaboración, haciendo uso de la ToolBox de CBGProcess, se podrá crear el modelo de proceso colaborativo, como detalla la figura C.5.

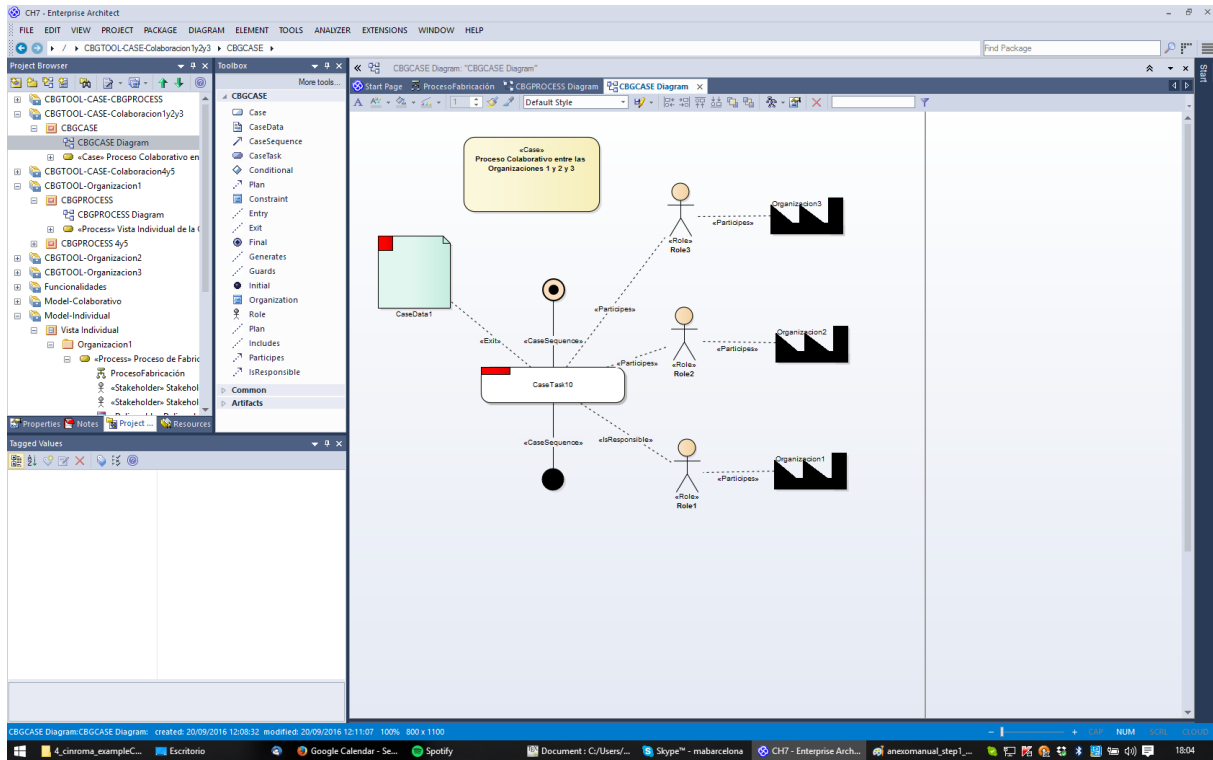


Figura C.5: Uso de CBGTool para crear el modelo de proceso colaborativo

C.4. Crear el modelo de caso colaborativo

Para pasar del dominio de procesos al de casos, procederemos de forma similar y gracias al uso de CBG-Tool podremos obtener un modelo de caso equivalente manteniendo la trazabilidad entre los elementos del modelo. Partiremos de un diagrama de proceso colaborativo conforme a CBGProcess y, mediante el menú contextual, seleccionaremos la extensión CBG-Tool y su acción *M2M CBGProcess-CBGCCase*. La figura C.6 expone gráficamente este escenario.

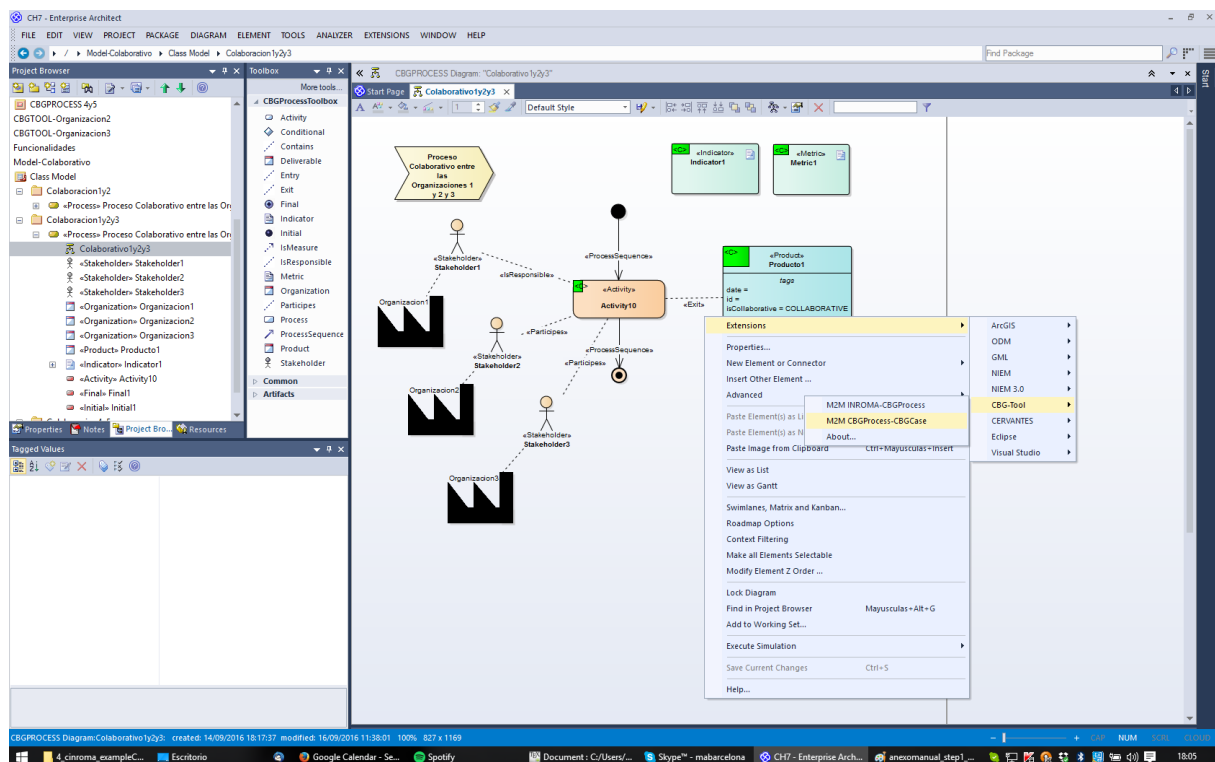


Figura C.6: Uso de CBGTool para transformar el modelo de proceso CBGProcess a CBGCCase

De forma automática el sistema generará el modelo de caso colaborativo equivalente conforme a CBGCase. A través de la ToolBox CBGCase incluida en el perfil UML desplegado en la solución CBG-Tool, el ingeniero de casos de la organización podrá editar el modelo de caso, como expone la figura C.7.

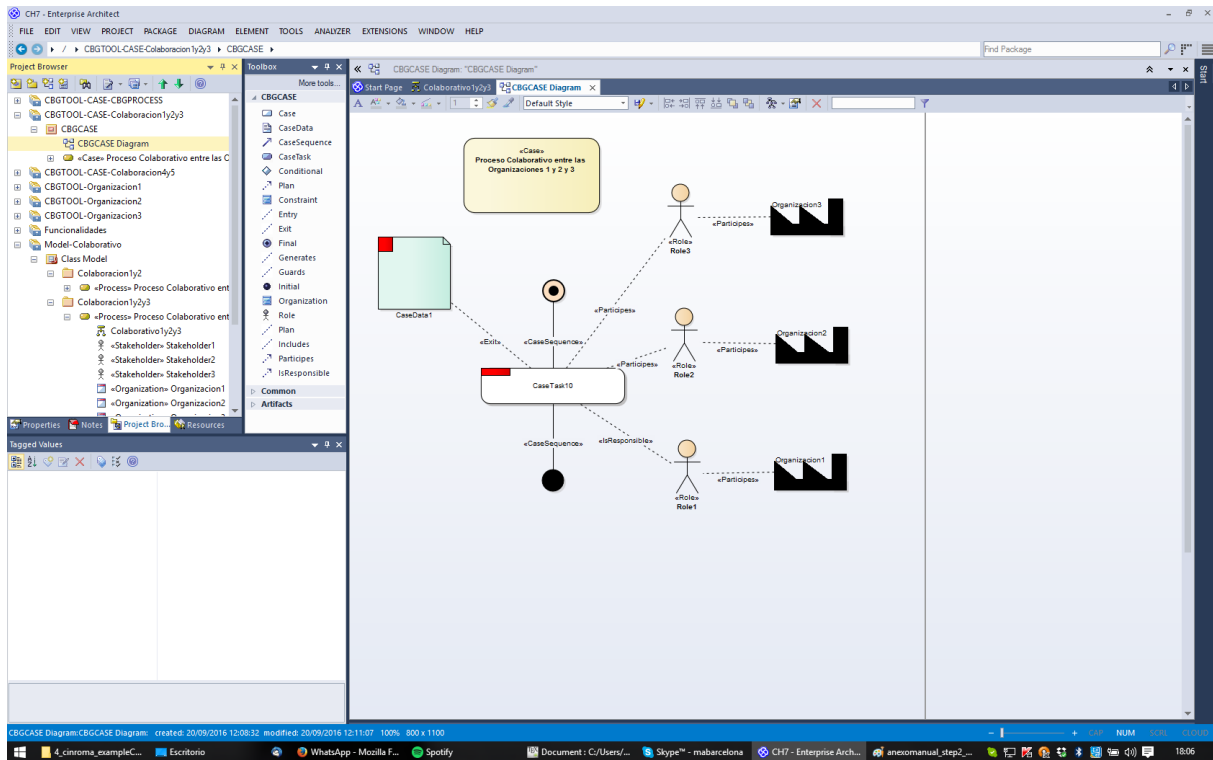


Figura C.7: Uso de CBGTool para crear el modelo de caso colaborativo

Por último, cada organización podrá, manteniendo la trazabilidad de los elementos, crear su vista individual del modelo de caso colaborativo, de nuevo haciendo uso de la ToolBox CBGCase, como se detalla en la figura C.8.

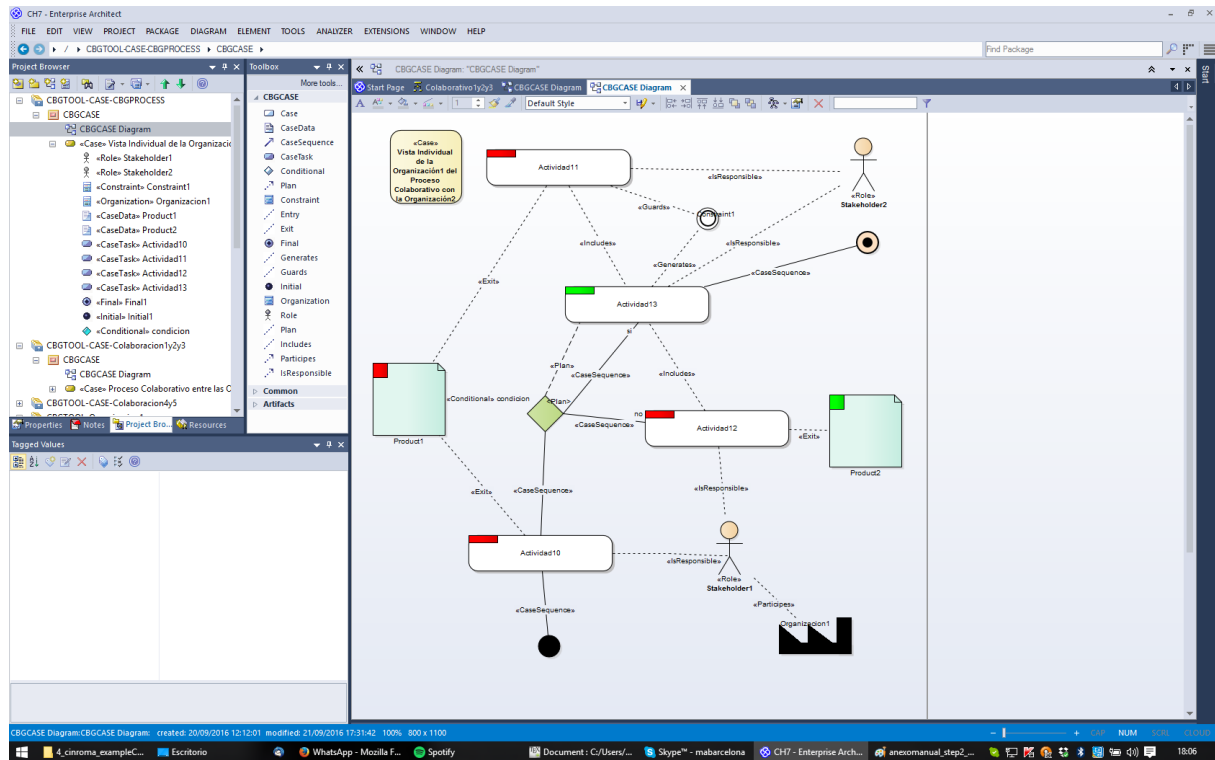


Figura C.8: Uso de CBGTool para crear la vista individual del modelo de caso colaborativo

Anexo D

Actividad investigadora

Este anexo recoge la producción y la actividad investigadora del doctorando. En la primera sección se recopila la información relacionada con la producción investigadora, esto es, capítulos de libros, artículos en revistas y artículos publicados en conferencias internacionales y nacionales.

En un segundo apartado se reúne la información referente a la actividad investigadora, es decir, estancias de investigación, comités, proyectos de I+D y redes de investigación en los que el doctorando ha colaborado.

D.1. Producción investigadora

D.1.1. Capítulos de libro

1. García-Borgoñón, L., Barcelona, M. A., Calvo, J. I., Ramos, I., Escalona, M. J. *CERVANTES: A Model-Based Approach for Service-Oriented Systems Development*. In Strahonja V., Vrcek N., Plantak-Vukovac D., Barry C., Lang M., Linger H., Schneider C. (Eds). Information Systems Development: Transforming Organisations and Society through Information Systems. Proceedings of the 23rd International Conference on Information Systems Development (ISD 2014), pp. 298-305. ISBN 978-953-6071-43-2. 2014.
2. García-Borgoñón, L., Blanco, R., García-García, J. A., Barcelona, M. A. *Applying Testing Techniques to Software Process Assessment: A model-based perspective*. In Escalona M.J., Aragón G., Linger H., Lang M., Barry C., Schneider C. (Eds). Information System Development: Improving Enterprise Communication. Proceedings of the 22nd International Conference on Information Systems Development (ISD 2013), pp. 333-344. ISBN 978-3-319-07214-2. 2014.
3. García-Borgoñón, L., Barcelona, M. A., Peña, P., Escalona, M. J. *SoftAragón: a methodological framework for Software Process Improvement in SMEs*. In Georgiadou P., Linecar P., Ross M., Staples G. (Eds). Software Quality in Flexible Development Environments. Proceedings of the 22nd BCS Quality Specialist Group's Annual International Software Quality Management (SQM 2014) pp. xx-xx. ISBN: 978-0-9926958-1-1. 2014.
4. García M.T., Barcelona, M. A., Ruiz M., García-Borgoñón L, Ramos I. *A Discrete-Event Simulation Metamodel for obtaining Simulation Models from Business Process Models*. In Escalona M.J., Aragón G., Linger H., Lang M., Barry C., Schneider C. (Eds). Information System Development: Improving Enterprise Communication. Proceedings of the 22nd International Conference on Information Systems Development (ISD 2013), pp. 307-317. ISBN 978-3-319-07214-2. 2014.
5. García-García, J. A., Victorio, J., García-Borgoñón, L., Barcelona, M. A., Domínguez-Mayo, F. J., Escalona, M. J. *A Formal Demonstration of NDT-Quality: A Tool for Measuring the Quality using NDT Methodology*. In Georgiadou, E., Ross M., Staples G. (Eds). Proceedings of the 21st BCS Quality Specialist Group's Annual International Software Quality Management (SQM 2013) pp. xx-xx. ISBN: 978-0-9563140-8-6. 2013.

D.1.2. Revistas

1. Barcelona M.A., García-Borgoñón L., López-Nicolás G. *Practical experiences in the usage of MIDAS in the logistics domain*. International Journal on Software Tools for Technology Transfer, pp. 1-15. 2016.
2. Barcelona M.A., *ICT for a Logistics Demonstration Centre*. ERCIM News number 105. 2016.

3. Barcelona M.A., García-Borgoñón L., López-Nicolás G., Ramos I., Escalona M. J. *Applying a model-based methodology to develop web-based systems of systems*. Journal of Web Engineering (JWE), *Under Review*, 2016.
4. Barcelona M.A., García-Borgoñón L., Mejias M., Ramos I., Escalona M. J. *Supply Chain Business Process Modeling Languages: a systematic literature review*. Simulation Modelling Practice and Theory, *Under Review*, 2016.
5. García-Borgoñón, L., Barcelona, M. A., García-García, J. A., Alba, M., Escalona, M. J. *Software Process Modeling Languages: A Systematic Literature Review*. Information and Software Technology (IST). Vol. 56, No. 2, pp. 103-116. 2014.
6. García-Borgoñón, L., Barcelona, M. A., García-García, J. A., Escalona, M. J. *Software Process Accessibility in Practice: A Case Study*. Procedia Computer Science (Proceedings of the 5th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion, DSAI 2013) . Vol. 27, pp. 292-301. 2014.

D.1.1.3. Conferencias internacionales

1. Barcelona M.A., López-Nicolás G., Meléndez E., Callao F. *An approach to include Web-based Interaction into Systems Of Systems*. In Majchrzak T.A., Traverso P., Monfort V., Krempels K-H. (Eds). Proceedings of the International Conference on Web Information Systems and Technologies (WEBIST 2016), , Rome, Italy, April 23-25, pp. 336-341. ISBN 978-989-758-186-1. 2016.
2. García-Borgoñón L., Barcelona M. A., Calvo J. I., Ramos I., Escalona M. J. *CERVANTES: A Model-Based Approach for Service-Oriented Systems Development*. Proceedings of the 23rd International Conference on Information Systems Development (ISD 2014), Zagreb, Croatia, September 2-4, pp. 298-305. ISBN 978-953-6071-43-2. 2014.
3. García-Borgoñón L., Barcelona M. A., Peña P., Escalona M. J. *SoftAragón: a methodological framework for Software Process Improvement in SMEs*. Proceedings of the 22nd BCS Quality Specialist Group's Annual International Software Quality Management (SQM 2014),, Southampton, UK, June 14, pp. xx-xx. ISBN: 978-0-9926958-1-1. 2014.
4. García-Borgoñón L., Barcelona M.A., García-García J.A., Escalona M.J. *Software Process Accessibility in practice: a case study*. 5th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion (DSAI 2013), Vigo, Spain, November 13-15, 2013.
5. García M.T., Barcelona M. A., Ruiz M., García-Borgoñón L., Ramos I. *A Discrete-Event Simulation Metamodel for obtaining Simulation Models from Business Process Models*. Proceedings of the 22nd International Conference on Information Systems Development (ISD 2013), Sevilla, Spain, September 2-4, pp. 307-317. ISBN 978-3-319-07214-2. 2014.
6. García-Borgoñón L., Blanco R., García-García J.A., Barcelona M.A. *Applying testing techniques to software process assessment: A model-based perspective*. Proceedings of the 22nd International Conference on Information Systems Development (ISD 2013), Sevilla, Spain, September 2-4, pp. 333-344. ISBN 978-3-319-07214-2. 2014.

7. García-García J. A., Victorio J., García-Borgoñón L., Barcelona M. A., Domínguez-Mayo, F. J., Escalona, M. J. *A Formal Demonstration of NDT-Quality: A Tool for Measuring the Quality using NDT Methodology*. Proceedings of the 21st BCS Quality Specialist Group's Annual International Software Quality Management (SQM 2013), London, UK, September 4, pp. xx-xx. ISBN: 978-0-9563140-8-6. 2013.

D.1.4. Conferencias nacionales

1. Barcelona M.A., García-Borgoñón L., Calvo J.I., Escalona M.J. *CERVANTES: Un framework para el diseño y desarrollo de sistemas distribuidos*. In Tuya J., Ruiz M., Hurtado N. (Eds). Proceedings of the 19th Jornadas de Ingeniería del Software y Bases de Datos (JISBD 2014), Cádiz, Spain, September 17-19, pp. 303-306, ISBN: 978-84-697-1152-1. 2014.
2. Guessous M., Barcelona M.A., García-Borgoñón L., Alba M. *Un Enfoque Basado en Modelos para incorporar Requisitos No Funcionales y de Integración de Software en el Diseño de Arquitecturas Orientadas a Servicios*. In Tuya J., Ruiz M., Hurtado N. (Eds). Proceedings of the 19th Jornadas de Ingeniería del Software y Bases de Datos (JISBD 2014), Cádiz, Spain, September 17-19, pp. 349-354, ISBN: 978-84-697-1152-1. 2014.
3. Barcelona M.A., Escalona M.J., Ramos I. *Un Enfoque Basado en Modelos para la Toma de Decisiones Colaborativa en la Cadena de Suministro*. In Moreno A., Vara J.M. (Eds). Proceedings of the 18th Jornadas de Ingeniería del Software y Bases de Datos (JISBD 2013), Madrid, Spain, September 17-20, pp. 262-267. ISBN: 978-84-695-8310-4. 2013.

D.2. Actividad investigadora

D.2.1. Estancias de investigación

1. Agosto 2013. Una semana de estancia de investigación en el Laboratorio de Investigación y Formación en Informática Avanzada de la Universidad Nacional de La Plata, Argentina, bajo la supervisión de Dr. Gustavo Rossi.
2. Julio-Agosto 2013. Seis semanas de estancia de investigación en la Southampton Solent University, Maritime and Technology Faculty, bajo la supervisión de Professor Margaret Ross.

D.2.2. Comités

- Miembro del Comité Organizador del 38th International Conference on Application and Theory of Petri Nets and Concurrency (PN2017).
- Miembro del Comité Organizador del 14th IFIP International Conference on Product Lifecycle Management (PLM2017).
- Miembro del Comité Organizador del 14th International Conference on Practical Applications of Agents and Multi-Agent Systems (PAAMS2016).
- Miembro del Comité de Programa del VIII Congreso Internacional de Computación y Telecomunicaciones (COMTEL2016) que se celebró en Lima (Perú) en Septiembre de 2016.
- Miembro del Comité de Programa del VII Congreso Internacional de Computación y Telecomunicaciones (COMTEL2015) que se celebró en Lima (Perú) en Septiembre de 2015.
- Miembro del Comité de Programa de la 23rd International Conference on Information Systems Development (ISD2014) celebrada en Zagreb (Croacia) en Septiembre de 2014.
- Miembro del Comité de Programa del VI Congreso Internacional de Computación y Telecomunicaciones (COMTEL2014) que se celebró en Lima (Perú) en Octubre de 2014.
- Miembro del Comité de Programa y del Comité de Organización de la 22nd International Conference on Information Systems Development (ISD2013), que se llevó a cabo en Sevilla en Septiembre de 2013.
- Miembro del Comité Organizador del 6th European Conference on ICT for Transport Logistics (ECITL2013).
- Miembro del Comité de Programa del V Congreso Internacional de Computación y Telecomunicaciones (COMTEL2013) que se celebró en Lima (Perú) en Octubre de 2013.

D.2.3. Redes de investigación

■ Redes nacionales

- Red CaSA - Calidad del software aplicada (TIN2010-12312-E9).
- Grupo de Calidad de Software - INES
- Red Temática para el desarrollo de soluciones de software de calidad en entornos PLM. Redes de Excelencia. Convocatoria 2015. (TIN2015-71938-REDT).

■ Redes internacionales

- AST - Red Internacional sobre arquitecturas de Testing.
- Red Temática Mexicana en Ingeniería del Software (244467) del Consejo Nacional de Ciencia y Tecnología de México en la convocatoria Conacit Redes Tematicas 2014.

D.2.4. Proyectos

Proyecto Empower	
Información del proyecto	Responsable: Julián Alberto García García, José Ponce Tipo de proyecto: Contrato 68/83 Referencia: P047-14/E09 Fecha de comienzo: 01/03/2015 Fecha de finalización: 31/12/2016

Red Temática para el desarrollo de soluciones software de calidad en entornos PLM	
Información del proyecto	Responsable: María José Escalona Cuaresma Tipo de proyecto: Plan Estatal 2013-2016 Excelencia - Redes Referencia: TIN2015-71938-REDT Fecha de comienzo: 01-12-2015 Fecha de finalización: 30-11-2017

Plataforma para la Modelización, Orquestación, Ejecución y Monitorización de Modelos de Proceso de Negocio	
Información del proyecto	Responsable: Francisco José Domínguez Mayo Tipo de proyecto: Contrato 68/83 Referencia: P047-14/E09 Fecha de comienzo: 25-11-2014 Fecha de finalización: 24-11-2016

Red Temática Mexicana en Ingeniería del Software	
Información del proyecto	Responsable: María José Escalona Cuaresma Tipo de proyecto: Consejo Nacional de Ciencia y Tecnología, CONACIT redes temáticas Referencia: 0244467 Fecha de comienzo: 01-12-2014
Financiado por	Consejo Nacional de Ciencia y Tecnología, México

Mecanismos Guiados en Etapas Tempranas para la Mejora del Software. Megus	
Información del proyecto	Responsable: María José Escalona Cuaresma Tipo de proyecto: Ministerio de Economía y Competitividad Referencia: TIN2013-46928-C3-3-R Fecha de comienzo: 01/01/2014 Fecha de finalización: 31/12/2016
Financiado por	Ministerio de Economía y Competitividad

Explorando Soluciones Guiadas para sistematizar el Aseguramiento Temprano de la Calidad del Software. Pololas	
Información del proyecto	Responsable: María José Escalona Cuaresma Tipo de proyecto: Ministerio de Economía y Competitividad Referencia: TIN2016-76956-C3-2-R Fecha de comienzo: 01/12/2016 Fecha de finalización: 31/12/2019
Financiado por	Ministerio de Economía y Competitividad

Testing Temprano y Modelos de Simulación Híbrida en Producción de Software. Tempros	
Información del proyecto	Responsable: María José Escalona Cuaresma Tipo de proyecto: Ministerio de Ciencia e Innovación Referencia: TIN2010-20057-C03-02 Fecha de comienzo: 01/01/2011 Fecha de finalización: 31/12/2013
Financiado por	Ministerio de Ciencia e Innovación

MIDAS: Model and Inference Driven Automated Testing of Services Architectures	
Información del proyecto	Tipo de proyecto: FP7-ICT Strep Referencia: FP7-ICT-2012-8-318786 Fecha de comienzo: 01/09/2012 Fecha de finalización: 31/10/2015
Financiado por	European Union

Red sobre Calidad del Software Aplicada. Red CaSA	
Información del proyecto	Responsable: María José Escalona Cuaresma Tipo de proyecto: Ministerio de Ciencia e Innovación Referencia: TIN2010-12312-E Fecha de comienzo: 21/07/2011 Fecha de finalización: 20/07/2012
Financiado por	Ministerio de Ciencia e Innovación

Testing temprano y modelos de simulación híbrida en la producción de software	
Información del proyecto	Responsable: María José Escalona Cuaresma Tipo de proyecto: Plan Nacional del 2010 Referencia: TIN2010-20057-C03-02 Fecha de comienzo: 01-01-2011 Fecha de finalización: 31-12-2013
Financiado por	Ministerio de Ciencia e Innovación

ITChain: Tecnologías de la Información para la Cadena de Suministro Colaborativa	
Información del proyecto	Responsable: Miguel Ángel Barcelona Liédana Tipo de proyecto: Plan AVANZA Referencia: TSI-020302-2010-80 Fecha de comienzo: 01/09/2010 Fecha de finalización: 29/03/2013
Financiado por	Ministerio de Industria

NOVARED: Sistema para distribuir e incrementar el valor creado en internet	
Información del proyecto	Tipo de proyecto: Plan Nacional de Investigación Científica, Desarrollo e Innovación Tecnológica Referencia: IPT-2011-1942-430000 Fecha de comienzo: 01/01/2008 Fecha de finalización: 31/12/2010
Financiado por	Ministerio de Economía y Competitividad

LOGysTIC-A: Centro Nacional de Conocimiento en la Aplicación de las Tecnologías de la Información y las Comunicaciones a la resolución de problemas en el ámbito de la Logística	
Información del proyecto	Tipo de proyecto: Plan AVANZA Centros de Conocimiento Referencia: TSI-070200-2008-76 Fecha de comienzo: 01/01/2008 Fecha de finalización: 31/12/2010
Financiado por	Ministerio de Industria

DBE: Digital Business Ecosystem	
Información del proyecto	Tipo de proyecto: FP7-ICT Strep Referencia: IST-2002-507953 Fecha de comienzo: 03/11/2003 Fecha de finalización: 31/01/2007
Financiado por	European Union

IST@HOME: Delivering video-based IST services into European HOMEs	
Información del proyecto	Tipo de proyecto: FP7-ICT Strep Referencia: IST-2000-28406 Fecha de comienzo: 01/02/2002 Fecha de finalización: 29/06/2004
Financiado por	European Union

VULCANO: Promoción del desarrollo de software libre en un entorno de calidad y confianza adaptando las metodologías, procesos, modelos de negocio y últimas tecnologías	
Información del proyecto	Tipo de proyecto: Plan AVANZA Referencia: TSI-020301-2009-1 Fecha de comienzo: 01/01/2007 Fecha de finalización: 01/01/2010
Financiado por	Ministerio de Industria

Acrónimos

ABMS Agent Based Modeling and Simulation

ACM Adaptive Case Management

ArchMDE Architecture-centric Model Driven Engineering

BPEL Business Process Execution Language

BPEL4WS Business Process Execution Language for Web Services

BPM Business Process Management

CBP Collaborative Business Process

CIM Computation Independent Model

CPFR Collaborative Planning, Forecasting and Replenishment

CPN Coloured Petri Net

CPPI Collaborative Process Performance Indicator

CR Continuous Replenishment

CRM Customer Relationship Management

CSM Customer Service Management

CSCMP Council of Supply Chain Management Professionals

DES Discrete Event Simulation

DFD Data Flow Diagram

DSVL Domain-Specific Visual Modelling Language

EIP Enterprise Integration Pattern

EML Enterprise Modelling Language

ESB Enterprise Service Bus

IDEF Integrated Definition for Function Modelling

ISO	International Organization for Standardization
ISM	Institute for Supply Management
IOBPM	Inter-Organizational Business Process Metamodel
IWT2	Ingeniería Web y Testing Temprano
KIP	Knowledge-Intensive Processes
MDA	Model Driven Architecture
MDE	Model Driven Engineering
MOF	Meta Object Facility
M2M	Model-to-Model Transformation
M2T	Model-to-Text Transformation
OCL	Object Constraint Language
OMG	Object Management Group
PAIS	Process-Aware Information Systems
PCI	Process Collaboration Information
PIM	Platform Independent Model
PIM4SOA	Platform-Independent Model for Service-Oriented Architecture
PPI	Process Performance Indicator
PSM	Platform Specific Model
PYMEs	Pequeñas y Medianas Empresas
QVT	Query View Transformation
RAD	Role Activity Diagram
RID	Role Interaction Diagram
RQ	Research Question
SC	Supply Chain
SCC	APICS Supply Chain Council
SCOR	Supply Chain Operations Reference
SCM	Supply Chain Management
SCMi	Supply Chain Management Institute
SCML	Supply Chain Modeling Language

SCML Supply Chain Markup Language
SCOL Supply Chain Ontology Language
SLR Systematic Literature Review
SOA Service-Oriented Architecture
SOC Service-Oriented Computing
SOE Service-Oriented Engineering
SPEM Software Process Engineering Metamodel
SRM Supplier Relationship Management
SysML System Modeling Language
SysPEM System Process Engineering Metamodel
UML Unified Modeling Language
UMM UN/CEFACT Modelling Methodology
VECCF Virtual Enterprise Chain Collaboration Framework
WfM Workflow Management
WSCI Web Service Choreography Interface
WSCL Web Services Conversation Language

Glosario de Términos

Capacidad de abstraccion (Abstraction) Capacidad de un lenguaje de aislar ciertos aspectos de los modelos de procesos concretos de forma que faciliten su comprensión

Capacidad de analisis (Analyzability) Capacidad de un lenguaje para disponer de los elementos necesarios para comparar diferentes modelos de procesos, tomar decisiones y actividades similares

Comprensible (Understandability)) Capacidad de un lenguaje de ser entendido por los usuarios del mismo

Comprobable (Testability) Capacidad de un lenguaje para realizar de forma automática pruebas que supervisen cómo se promulga el modelo

Ejecutabilidad (Executability) Capacidad de un lenguaje para soportar construcciones con semántica operacional que ofrezcan el soporte para la ejecución de esas construcciones

Escalabilidad (Scalability) Capacidad de un lenguaje para describir procesos de diferentes tamaños, tanto grandes como pequeños

Estandarizacion (Standardization) Grado de cercanía de un lenguaje a estándares o lenguajes de uso generalizado como UML

Expresividad (Expressiveness) Capacidad de un lenguaje para expresar lo que realmente se realiza durante los procesos de desarrollo de software

Flexibilidad (Flexibility) Capacidad de un lenguaje para poder ser aplicado en una variedad de entornos de procesos, no únicamente en el ámbito del software

Formalidad (Formality) Capacidad de un lenguaje para expresar formalmente la sintaxis y semántica

Gestion de Procesos de Negocio (Business Process Management) Conjunto de métodos, técnicas y herramientas que soportan el diseño, aprobación, gestión, análisis y mejora de los procesos de negocio

Ingenieria del Software Disciplina o área de la informática que ofrece métodos y técnicas para el desarrollo y el mantenimiento de calidad que resuelva todo tipo de problemas

Ingenieria orientada a modelos (Model Driven Development) Paradigma de desarrollo de software que se centra en la creación y explotación de modelos de dominio

Modelo de Proceso Descripción abstracta de un proceso actual o propuesto que representa elementos seleccionados del mismo, que son considerados importantes para el propósito del modelo y pueden ser ejecutados por una persona o una máquina

Modularidad (Modularity) Capacidad de un lenguaje para combinar diferentes pedazos o componentes de procesos con el objetivo de construir uno nuevo

Navigational Development Techniques Propuesta metodológica web guiada por modelos que ofrece un soporte completo, incluyendo herramientas, para todo el ciclo de vida de un sistema software: desarrollo, pruebas, aseguramiento de la calidad y gestión del proyecto

NDTQ-Framework Framework de trabajo que define todos los procesos actualmente sopor-tados por NDT

Proceso Consiste en el uso de los recursos para transformar entradas en salidas debido a que algún tipo de trabajo, actividad o función se ha llevado a cabo

Proceso de Negocio Conjunto de tareas relacionadas de manera lógica que se deben realizar para proporcionar valor a los clientes o cumplir otros objetivos estratégicos

Proceso de Software Conjunto de tecnologías, mejores prácticas y estrategias que permitan gestionar de manera efectiva el desarrollo realizado por grandes equipos de ingenieros software a lo largo de todo el ciclo de vida de los productos

Promulgable (Enactability) Capacidad de un lenguaje para que un modelo pueda imitar la ejecución real del proceso

Reflexion (Reflection) Capacidad de un lenguaje para soportar directamente la evolución de los modelos de procesos

Representacion grafica (Graphical representation) Capacidad de un lenguaje para disponer de una semántica gráfica que facilite su descripción e intercambio entre los usuarios del mismo

Simulable (Simulability) Capacidad de un lenguaje para que un modelo pueda ser simulado previo a su ejecución

Sintaxis Abstracta Describe la estructura del lenguaje y la forma en la que se pueden combinar los distintos elementos, independientemente de su representación

Sintaxis Concreta Especifica la representación legible (gráfica o textual) de los elementos abstractos

Soportado por herramientas (Tooling Support) Disponibilidad de herramientas que nos permitan definir los procesos con el lenguaje

Soporte a multiples vistas (Support Multiple Views) Capacidad de un lenguaje para disponer de los elementos semánticos necesarios para la definición de vistas desde varias perspectivas del modelo de procesos

Transformacion Mecanismo que nos permite derivar unos modelos a partir de otros modelos ya existentes

Transformacion Horizontal Aquella en la que el modelo derivado y el origen tienen el mismo nivel de abstracción

Transformacion M2M Transformación Modelo-a-Modelo: representa una relación entre dos sintaxis abstractas y se define mediante un conjunto de relaciones entre los elementos de los correspondientes metamodelos

Transformacion Vertical Aquella en la que el modelo derivado tiene un menor nivel de abstracción que el modelo inicial